SINCLAIR QL WORLD

NOVEMBER 1992 £1.95

# LONELY JOKER
## PUTTING HIS CARDS ON THE TABLE

# TRANS 24:
## BRYAN DAVIES PINS IT DOWN!

# WHERE HAS YOUR MEMORY GONE?

# DIY TOOLKIT

# QMON
## MONITOR & DEBUGGER

ISSN 0951-9335

# DJC

# Dilwyn Jones Computing

*41 Bro Emrys, Tal-y-Bont,*
*Bangor, Gwynedd LL57 3YT U.K.*
*Tel: Bangor (0248) 354023*

## QL SOFTWARE

NB PLEASE NOTE THE MEANING OF CODES IN SQUARE BRACKETS BELOW:
[R] RAMDISK REQUIRED
[F] AVAILABLE ON FLOPPY DISK
[M] AVAILABLE ON MICRODRIVE
(NB NOW NO EXTRA CHARGE FOR SUPPLYING SOFTWARE ON MICRODRIVE)
[128K/512K] MINIMUM MEMORY REQUIRED.
[PC] FOR IBM PC AND COMPATIBLES.
[TK2] TOOLKIT 2 REQUIRED.

## CATALOGUE

PLEASE ASK FOR A COPY OF OUR QL SOFTWARE CATALOGUE. WE NOW SELL SO MANY ITEMS THAT WE CANNOT PROVIDE MUCH DETAIL IN THE ADVERT. THE CATALOGUE CONTAINS MORE DETAILS OF THE PROGRAMS AND A SAMPLE SCREEN DUMP FROM SOME OF THEM. CALL FOR A COPY, OR ASK FOR A COPY WITH YOUR ORDER. USER GROUPS - ASK FOR A PACK OF CATALOGUES FOR YOUR MEMBERS AND TAKE ADVANTAGE OF THE DISCOUNTS AVAILABLE FOR BUYING SEVERAL PROGRAMS AT THE SAME TIME!

## FLOPPY DISKS & LABELS

| | |
|---|---|
| 3.5" DSDD UNBRANDED DISKS | £0.40 |
| 3.5" DSHD FLOPPY DISKS | £0.70 |

MICRODRIVE CARTRIDGES - PLEASE PHONE BEFORE ORDERING TO ASK ABOUT PRICE AND AVAILABILITY

| | |
|---|---|
| 100 DISK LABELS | £2.00 |
| 100 DISK LABELS ON PRINTER ROLL | £2.50 |
| 100 ADDRESS LABELS ON ROLL | £2.00 |
| 100 MICRODRIVE LABELS ON ROLL | £2.00 |

*Please add £0.50 postage per roll of labels (max postage £2.50) if only ordering labels. Add £2.50 per order for floppy disks. No extra postage on microdrive cartridges.*

## FILE TRANSFER

DISCOVER £20.00
*[F 256K] Copy files from QDOS to PC disks and vice versa. No cables required. Needs either a dual disk drive on the QL or single disk drive plus a ramdisk.*

MULTI DISCOVER £30.00
*[F 256k] Enhanced version, also transfers between QDOS and BBC micro DFS/ADFS formats, CPM and Unix CPIO formats. Needs either a dual disk drive on the QL or single disk drive plus a ramdisk.*

TEXTIDY £15.00
*Strip out control codes from Quill files and assist Discover with preparation of files for transfer to other computers.*

OPD INTERCHANGE £15.00
*File transfer utility for transferring data files and BASIC programs between microdrives on QL and ICL OPD micros. A quality program by the author of Discover.*

QL-PC FILESERVER £24.50
*[F 128k] NEW! The latest from Di-Ren! Allows transfer of files between QL and PC's connected via a simple serial link. Wiring details supplied, or ready made cables available from TF Services. File transfer is as simple as a COPY statement if required - PC_USE command allows files to be copied to a device called PCDA, PCDB (PC drive numbers) or to a PC's LPT port! Use standard QL file handling commands. This program even works on an unexpanded QL and includes software for both the QL and PC ends on disk.*

## LEISURE

SOLITAIRE £15.00
*[F M 128k] A very addictive patience (Klondyke) card game. Great fun, simple to use.*

CRICKET SECRETARY £12.00
*[F M 128k] Record, display and print cricket averages etc. For cricket buffs everywhere!*

QUESTION MASTER £10.00
*[F M 128k] Questions and answers, for revision or entertainment. Make your own or try our quizzes.*

QUIZ SETS each £5.00

FLEET TACTICAL COMMAND II QL £49.95
*[F 512k] Naval strategy game, playable on two networked QLs.*

FLEET TACTICAL COMMAND II PC £69.95
*[PC] Licenced PC version of this superb game. Networks to a QL via a simple serial port link (cable available from TF Services). Compatible with QL version, but some enhanced features such as improved graphics. Contact us for more details and PC requirements.*

COMBINED QL & PC VERSIONS £85.00
*Package deal for both QL and PC versions together. Users already with the QL version should contact Di-Ren for upgrade enquiries.*

FTC2 DATA PRINT UTILITY £9.95
*[F 128k] Print out data from FTCII described above.*

THE FUGITIVE £9.95
*[F 512k] Text adventure game.*

COCKTAILS WAITER £10.00
*[F 256k] Database of alcoholic drinks recipes, hundreds of recipes supplied, more available.*

RECIPE SETS each £5.00

## PROGRAMMING

S_EDIT £20.00
*[F 384k] Simple to use menu driven editor. Can be used to edit binary or plain text files, to edit or create assembler files, and so on. Block handling, word wrap, search and replace, dynamic linked list memory handling.*

EASYPTR II £49.00
*[F 256k] Pointer environment programming aid.*

EASYPTR II BUDGET VERSION £35.00

DISA DISASSEMBLER £29.00
*[F 256k] Interactive pointer driven disassembler.*

BASIC REPORTER £10.00
*[F M 128k] BASIC programming aid, list variables, etc.*

BUDGET QLIBERATOR COMPILER £25.00
*[F M 128k] BASIC compiler suitable for use on unexpanded QL.*

QLIBERATOR BASIC COMPILER £50.00
*[F 256k] More sophisticated version, excellent BASIC compiler for the QL.*

QLOAD & QREF UTILITY £15.00
*[F M 128k] QLOAD is a fast load utility for BASIC programs. QREF lists variables, procedures, etc used in BASIC programs.*

MEGA TOOLKIT on disk £25.00
on EPROM cartridge and disk £40.00
*Comprehensive toolkit of over 200 BASIC extensions which can be used in your BASIC programs. Can also be used in published software with no royalty payment. Covers a very wide range of subjects, ask for information. Large file of BASIC demonstration routines and comprehensive manual.*

## FILE HANDLING

LOCKSMITHE £14.95
*[M 128k] Microdrive backup program, byte for byte!*

4MATTER £23.50
*[F M 384k] Complete software backup system. Can transfer many microdrive only programs to disk and make safety backups of protected microdrive software.*

TOOLCHEST £14.95
*[M F 256k] Utilities and routines to allow creation of a customised microdrive doctor program, with on-line help manual. Can display microdrive information in hex and ASCII, etc.*

FILES 2 £12.00
*[F M 128k] Simple to use popup file handling utility for copying, deleting, renaming, viewing etc files. Easy to use but very powerful, everything done from menus.*

FILEMASTER £12.00
*[F M 512k R] For bulk copying of files, etc quickly via ramdisk. Also does simple disk labelling and files printout.*

THE GOPHER £12.00
*[F M 128k] File finder, searches through disk or microdrive for file containing given string.*

WINBACK £25.00
*[F 256k] Hard disk and ED disk backup utility for Miracles and compatibles. This is version 2, able to split very large files into smaller sections on the backup floppies and join them back together later.*

## DISPLAY SOFTWARE

BANTER £25.00
*[F 512k] Banner making program. Uses outline fonts for improved print quality and smooth edges to large letters. Eight fonts supplied. Easy to use, prints to 9 pin and 24 pin Epson compatible printers and to HR5 printers. Print sideways across up to 4 pages, with up to 5 lines of large text. We use it to make banners for our stand at shows, use for advertising, notices, shops, schools, exhibitions, etc.*

VISION MIXER 1 £10.00
*[F 512k] Screen display system, use as an eye catching screen display system with QL mode 4 or 8 screens. Version 1.5 now has built in speed control for Gold Card use. Upgrade £3.00*

VISION MIXER PLUS £22.50
*[F 384k] Enhanced, menu driven version.*

PICTUREMASTER £15.00
*[F 256k] Screen drawing utility for use with Vision Mixer, etc.*

PICTUREMASTER PLUS £20.00
*[F 384k] Enhanced version, extra features.*

UPGRADE TO "PM PLUS" £5.00

## PRINTER DUMPS

SIDEWINDER PLUS £24.95
*[F M 512k] Screen and desktop publisher page file printer dump utility. Print in various sizes, from very small postage stamp or label size to very large banner size which can be in strips, ready to stick together later. Print sideways, mirror, invert, mode 4, mode 8, grey shade, add text. Suitable for 9 and 24 pin printers, reconfigurable printer drivers. Prints Page Designer 2 and DTP pages. Can print to LC10 and JX80 colour printers.*

SINCLAIR QL WORLD

***Editor***
Helen Armstrong

***Publisher***
Mark Kasprowicz

***Advertising Manager***
Jim Peskett

***Creative Director***
John Stanley

***Graphic Artist***
Stevie Billington

***Magazine Services***
Linda Miller, Frances Maxwell,
Pauline Wakeling

**Sinclair QL World,**
**Published by Arcwind Ltd.**
**The Blue Barn,**
**Tew Lane, Wootton,**
**Woodstock,**
**Oxon. OX7 1HA**
**Tel: 0993 811181**
**Fax: 0993 811481**
**ISSN 026806X**

If you have any comments or difficulties please write to the editor and we will do our best to deal with your problem in the magazine, though we cannot guarantee individual replies.
Back issues are available from the publisher price £2.50 UK, £2.99 Europe. Overseas rates on request.

Subscriptions from: Arcwind The Blue Barn, Tew Lane, Wootton, Woodstock, Oxon. OX7 1HA
UK: £23.40
Europe: £32.90
Rest of World: £40.90

Reprographic Services: The Wace Group, London.
Distributed by: Seymour Press Ltd.,
Windsor House, 1270 London Road, Norbury, London, SW16 4DH



# Contents

November 1992



## Coming Soon

Competition - 60 disks of public domain software to win. Get your pens out!

# PERFECTION SPECIAL EDITION

## POWER

PERFECTION SPECIAL EDITION has 253 (two hundred and fifty three) direct/menu commands (not counting options in sub-menus), plus 32 special characters (like Bold on) that can be inserted 'directly' plus intelligent (and now excellently documented) macros. Comparisons with other word processors on the subject of power are hence quite unnecessary.

## EASE OF USE


Independent reports, customer feedback and published reviews (of its less able but still excellent predecessor, PERFECTION) leave one in no doubt as to which word processor is friendliest – PERFECTION SPECIAL EDITION, with its intuitive, silky handling. Uniquely, it has two operating modes, with both menus (visible or invisible – they even look like Quill's) and direct commands (for when you familiarise yourself with the system). Uniquely, both modes are 're-entrant' (so you can use any menu option or direct command while you are in the middle of performing another option or command – block handling, etc, becomes a dream). Uniquely, PERFECTION SE has fully automatic memory management, grabbing and releasing RAM instantly as your document grows or shrinks – programs without this don't take full advantage of the multi-tasking abilities of the QL! Uniquely, PERFECTION SE leaves you in the driving seat, not juggling things around 'underfoot' while you are typing. Uniquely, PERFECTION SE allows up to nine different documents to be handled simultaneously from one copy of the program – each with totally independent margin, tab, justification, control panel, etc, settings. Uniquely, each document can itself have up to six environment settings, each settable or recallable instantly with a single keypress combination. Each document can have any number (up to 500,000 on GOLD CARD) of candidate blocks! Each document can have two independent windows (of any depth, of any (but same) width across) 'on to' it, even with overlapping text – that allows you to edit in one place while viewing another, to compare 'before editing' with 'after editing' (you can arrange to have one window remain 'frozen' in time), etc. Uniquely, we realise how much faster it is to type in something like CTRL/SHIFT/F5 than (say) F3 D U – both involve three keys, but as the former doesn't require the keys to be pressed in just one specific order, or to be released in any order at all (together will do), it is in practice twice as fast as the latter, where no key may be pressed until its predecessor is released. Also, sequences like CTRL/T (top) and then CTRL/G (go to next occurrence of string in set direction) can be accomplished by holding down CTRL and then tapping T and G. Uniquely, by providing eight user-definable strips, PERFECTION SE allows you to cope with printers of the future, not just the printers that now exist – you can attach the strips to any printer features. Uniquely, PERFECTION SE's status lines give full information on all relevant global settings. And the manual has an index. Also, it has all the important bits at the front.

**PC CONQUEROR GOLD SPECIAL EDITION** – This terrific new product for QLs with 1.5 Mb or more makes your QL system into a PC. A well-equipped PC too, with about a megabyte of expanded RAM installed, and the ability to read, write and format SD/DD/HD/ED disks (the last by making them into pseudo hard disks). Disk performance is up to 5 times faster. Other performance is up to 55% faster than standard CONQUEROR on GOLD CARD. There are many extra features too – see our ads in June - September 1992 QLW for full details.

**DR-DOS V 6.0** – The latest and most capable DOS of all!

**QMATHS MATHEMATICAL SYSTEM PART TWO** – An excellent complement to QMATHS, with loads of 'functionality' – fractals, function evaluation, terrain plotting, masses of maths & stats, etc.

**QUICKLASER** – The definitive output tool from PRO PUBLISHER to HP LaserJet II (or compatible) printers. Printed output quality subjectively exceeds that from any other QL product.

**TRANSFER UTILITY SPECIAL EDITION** – Does everything – 16 case change options, 14 types of sorting (multiple sorts possible), auto string translations, etc.

**LIGHTNING SPECIAL EDITION GOLD CARD VERSION** – See June-Aug 1992 QLW for details: optimal speed from GOLD CARD, ST/QL, THOR XVI. Free upgrade from the ROM SE version (return ROM + disk) if you are ordering something else at same time: if not, £10 charge.

## SUPERB PRINT QUALITY & FLEXIBILITY

Uniquely, using the aforementioned automatic link, you can output PERFECTION SE documents using over a thousand fonts (a huge variety of styles and sizes, supplied on the PUBLISHER and TOOLBOX disks) on virtually any printer – from the humblest Epson RX80, Brother M1009 or Star LC10 (which are all single font machines when used with most word processors) to top-end lasers. *You are not limited to the fonts built into the printer!!* All PERFECTION SE **bold**/underlined/ *italics*/$^{super}$/$_{sub}$, etc, settings are preserved. Proportional spacing and micro-justification are automatic, even when you mix fonts of differing widths *and heights* (even on the same line), vary line spacings, etc.

Uniquely, you are not trapped with one type of micro-justification (ie adding all the space between words, and using the predefined widths of characters as their separation) – with our system, you can vary (in 5% steps) the ratio of micro-spaces added between words to that added between characters (the latter in proportion to their *individual* widths). Ratios around 65%-35% – not the 100%-0% forced upon you by some other word processors – seem to give the most pleasing results. Uniquely, you are not limited to mere rectangular columns plus headers/footers – that's all the rest can do – you can output in any sequence to any number of frames (text flowing from one to the next), each of any shape – irregular polygons of up to 66 sides, circles, multi-column or part-column boxes (hundreds of types of borders, thousands of textures), doughnuts, wrap-around shapes, even re-entrant ones ('join-the-dots' type borders, even with intersecting edges) – all with micro-justification and proportional spacing! Look at the example on this page. Of course, if super-fancy effects (like wraparound windows and mixing different font widths on the same line while maintaining right justification) are not of the essence, PERFECTION SE's direct printer output is excellent with all your printer's capabilities supported.

## THE FASTEST

For benchmarking, we've used a public domain version of the first book of The King James Bible, all fifty chapters of the book of Genesis. This came to **one hundred and forty pages**, well over **forty two thousand words** excluding headers and footers, well over **two hundred and twelve thousand characters** excluding justification ones, **fifty full chapters** and **one thousand five hundred and thirty three indexed verses**!! We didn't use a smaller file (as used to benchmark other programs) as PERFECTION SE's timings for most operations then become impossible to stopwatch (too fast!). The hardware used for all timings was GOLD CARD: speeds would be **further improved by over three times** using the ST/QL 030. Of course, LIGHTNING SE was used. File operations were to ramdisk: normal slave blocks would give identical times. All settings on **everything** were for maximum speed, except where indicated to the contrary – we do not force full speed upon you in operations like scrolling and global Search & Replace. PERFECTION SE's speed for these is switchable (at run-time and when configuring), as too great a speed may cause overshoot (with scrolling) or fatal alteration (if there is human error inputting the target or replace strings). Here are the benchmarks for this huge document:

Load 140 pages: 0.6 seconds (yes 0.6, not 6!) ☆ Import 140 pages: 0.6 seconds (yes 0.6, not 6!) ☆ Save 140 pages: 0.5 seconds (yes 0.5, not 5!) ☆ Export 140 pages: 0.5 seconds (yes 0.5, not 5!) ☆ Case-sensitive search from top for word at bottom: 0.4 seconds (yes 0.4, not 4!) ☆ The same, but case case-insensitive: 0.5 seconds (yes 0.5, not 5!) ☆ Case-sensitive search backwards from bottom for word at top: 0.4 seconds (yes 0.4, not 4!) ☆ The same, but case-insensitive: 0.5 seconds (yes 0.5, not 5!) ☆ Automatic Search & Replace, in Fast (No Query) mode, of last 600 occurrences: 7.4 seconds (same length replace string); 7.7 seconds (shorter replace string); 10.5 seconds (longer replace string – longer time as we deliberately chose a high *density* of replaces to handicap PERFECTION SE into auto-managing memory – without causing any heap fragmentation, but still with only a 0.005 second overhead per replace!) ☆ Automatic Search & Replace in Slow ('Querying') mode: arbitrarily slow, typically 30 times slower – because we deliberately allow for human response time (in case you want to abort) before proceeding from one replace to the next. ☆ Scrolling 100 lines of text, up or down, by full-width screen page: 1.5 seconds ☆ Scrolling 100 lines of text on full-width screen, line by line, in slow (full) mode: 5.7 seconds (down)/5.8 seconds (up) ☆ As above, but in medium speed mode: 4 seconds ☆ The same, but in fast mode and default settings: 13.5 seconds to scroll through the whole massive document, averaging 0.23 seconds per 100 pages (!) – and this could be made up to ten times faster by reconfiguring PERFECTION SE ☆ Reformatting paragraphs, changing margins, justification, etc, of existing text: c5 times faster than predecessor ☆ Inserting (or undoing) emphasised, underlined, italics, superscript, subscript, 8 strips, 6 environment settings: Instant (i.e. immeasurable) ☆ Navigation to line or page or to top or bottom or to 8 markers or to highlights/blocks: Instant ☆ Setting new margins, justification, etc: Instant ☆ Deleting block of 100 pages: 0.3 (yes, 0.3 not 3!) seconds ☆ Copying/moving block of 100 pages (not just 10!), downwards or upwards: 3.4 seconds (yes, including all the time for automatic memory management and anti-fragmentation – other programs are light-years behind) ☆ Spellcheck as you type: Ten times faster than anyone can possibly type ☆ Spellcheck all 140 pages in the document using the 350,000 word Mega Dictionary: 3.9 seconds (20 'errors' – like 'pluckt'!) ☆ And using our tiny dictionary (well, tiny by our standards – large by comparison with most others): 5.1 seconds (566 'errors') ☆ Time taken to create user dictionary from the results of the second spellcheck (566 errors): 0.8 seconds to extract all 'errors' from document and clean document; 1.9 seconds to create a full user dictionary therefrom and also a sorted, duplicate-free wordlist file (for browsing) ☆ Spellcheck file (ASCII or native): Even faster. ☆ Print first 10 pages to file: 3.5 seconds. ☆ Change every occurrence in 140 pages of God to ***God*** in bold underlined italics, strip 8 – 9.5 seconds! ☆ Virtually everything else: instant.

For prices, see the coupon page of our ad. For more info, read our detailed QLW ads in early 1991 for PERFECTION, plus the extra features of the SE (well, about half of them) listed in the June-August 1992 issues. You can upgrade from the standard PERFECTION (or PLUS) to the SPECIAL EDITIONs for the difference in current price, plus £10 (no manuals or dictionary disks to be returned – we'll send a supplement to the manual.

## PERFECTION
## PERFECTION PLUS

Perfection is the finest word processor available for any computer. We have received dozens of letters from happy users saying just this... and all of these letters were unsolicited. "Superb" was used most often.

Perfection manages to achieve all the sophistication of the most complex PC word processors while still using a user interface as friendly as Quill's. Perfection has a dual system of user control: menus while you are familiarising yourself with the program, and direct commands for the time when you feel ready for more adventurous things. The two systems can be used interchangeably and even simultaneously. Even more exciting – both systems are iterative. In case you don't understand what this means, let us give you an example: suppose you wished to move a block of text using the menus. You would choose Block Move (yes, it is right in the first menu) and the screen would then tell you to move your cursor to the start of the block. On most word processors you would have to navigate manually to this position: indeed, on many of them (Quill included) only a subset of the normal navigation commands would be available. On Perfection, not only can you use all the manual navigation commands (viz all 28 permutations of CTRL, ALT, SHIFT and the arrow keys!) but in addition you can use direct commands like GoTo Line or Page or any of eight markers. Even more amazingly, you can use Search (either as a direct command or from the menus) *even though you are already 'within' a menu option.*

Perfection has about 200 commands, but the layout of menus and the choice of keys for the direct commands makes it very easy to master. Though a 100+ page manual is provided (with all the important bits right at the front), you should only need to consult it for specialised operations like macros.

Even if speed is not particularly important to you, we assure you that Perfection's lightning performance will enable you to use the word processor in sensible ways that you would not have dreamed possible before. For example, scrolling 100 pages or so is accomplished so quickly using the normal navigation commands that you do not need to bother using a menu option to do the move. Spellchecking, assuming you have Perfection Plus, is accomplished virtually instantly: to spellcheck this whole ad (all the pages) would take under 1.5 seconds... Searching (you can switch case sensitivity, as well as equivalences between tabs, soft spaces and hard spaces) is at the rate of about 100 A4 pages per second.

Moving from one word processor to another is usually very traumatic. With Perfection, this will not be the case. Not only can Perfection read in Quill _doc and _exp files directly (you do not even need to tell it they are Quill files!) but it can make direct and immediate use of your existing Quill printer driver. File re-export is also possible.

Perfection is truly WYSIWYG: this means that bold appears bold on screen, italics appear as italics, underlined as underlined, and so on. Of course, your printer may have functions we do not know about (upside down?). To deal with these, Perfection provides a number of on-screen shaded strips: these can be attached to any printer function you wish, and will not upset justification as a translate would. Of course, translates are provided as well!

A variety of statistics on the document being processed are available: some of them are on view all the time, the rest can be toggled to instantly. Not only is there a word count, but also page, line, character and special character (like Superscript Off) counts. There are also a dozen status indicators, letting you know whether you are in Insert or Overwrite mode, whether a block is defined, whether interactive spellchecking is enabled etc. Current line (from top as well as within page) and column positions and character codes are also available.

A terrific feature of Perfection is the dual screen mode. You can view one part of the document while editing another. The sizes of the two windows are themselves adjustable, both in real-time or via the configurator. We should devote more space to the configurator: however, it must suffice to say that everything that could be dynamically set within Perfection may also be preset with the configurator. The configurator can, for example, allow you to select any of 256 colours for any of a dozen parameters (like paper colour, border colour, status window ink and paper colour etc).

Perfection is fully multitasking without need for any external accessory: however, if you already use QPAC or Taskmaster or similar and are happy, you may go on doing so.

There is absolutely no way that we can prepare you for the quality 'feel' of Perfection. We have a great deal of experience using PC word processors costing many hundreds of pounds: with absolutely no exception, Perfection is far easier to use and master.

So if you thought Perfection was unattainable, you have a very pleasant surprise coming to you!

## LIGHTNING SPECIAL EDITION
## LIGHTNING

These programs accelerate QL operation by up to 10x (2x –4x is typical) without having any adverse effect whatsoever on compatibility or anything else. Lightning SE is typically 40% faster than the standard version. This acceleration is totally independent of, and in addition to, any speed-up obtained by hardware means. So if you have Gold Card, your need for Lightning SE is just the same as if you had only an unexpanded QL – Lightning SE will accelerate both by the same ratio.

The Lightning programs achieve their acceleration by automatically paging out sections of the QL's operating system and replacing these with optimal, concise code written by us.

Lightning installation is a completely automatic and one-off: no knowledge of computing or programming is required. Once installed, Lightning can be completely forgotten about – you will soon get used to the superb speed! Knob twiddlers are catered for too.

Lightning technology is not built in to any of our other programs. Perfection users (as well as users of all other QL software) should therefore use Lightning all the time.

In summary: if you do not have Lightning, you are wrong. Buy this one **FIRST OF ALL!**

## PROFESSIONAL PUBLISHER

The Professional in Professional Publisher refers to the quality of output from that program, and is not meant to suggest any complexity of operation. Few programs are as easy to use as this one: > 99% of users will be able to do without using a manual!

DIGITAL PRECISION LTD

Professional Publisher is by far the best DTP program for the QL. It is fully compatible with Perfection, Editor, Quill, Eye-Q & the ASCII editors. It allows you to *both* create and import *both* text and graphics. Text can be 'poured' into boxes of any shape , size and number, automatically maintaining justification and hyphenation settings. So flowing text around graphics is a doddle.

Professional Publisher is supplied with a generous selection of fonts of various sizes, as well as clip art.

Justification is by pixel, not by character. This gives a much smoother effect.

It is pointless for us to try to list all of Professional Publisher's features – we would end up filling half the magazine! We will concentrate on just a few 'points': Professional Publisher is extremely precise, performing all its computations accurate to a small fraction of a millimetre. All its features can be preset by you using its configurator, ruling out the need for repetitive key strokes.

The program is extraordinarily versatile while remaining intuitive in its user interface. Buy it!

## PROFESSIONAL PUBLISHER
## TOOLBOXES

Toolbox I is an excellent collection of high definition fonts, clip art and utility programs for Professional Publisher. While the fonts supplied with Professional Publisher are excellent, many users will feel the need for a wider range of typefaces and styles.

Toolbox II starts where Toolbox I leaves off, providing an even better – and different – font collection.

The two Toolboxes complement each other and are available together at a special price.

## FONT ENLARGER
## GRAFIX

Font Enlarger does exactly what you would expect it to from its name. While Professional Publisher is also capable of enlarging fonts, it does them 'on the fly' and consequently is not able to remove the jaggedness caused by magnification. Font Enlarger is much cleverer, and enhances detail without any step effect.

While the built-in printer driver for Professional Publisher is excellent with 9-pin printers, it is not optimal with 24-pin or laser printers. Grafix is.

DIGITAL PRECISION LTD

## EYE-Q
## ULTRAPRINT

Eye-Q is the finest graphics program for the QL. While there may be other graphics programs with a few more features, no other program comes anywhere close to Eye-Q in sheer enjoyability. Eye-Q develops a pleasurable tactile relationship with you, and makes you feel like an artist (even if you aren't). Eye-Q graphics can be read in by Professional Publisher, and the latter's pages can be exported to Eye-Q (using Toolbox I). Everything in Eye-Q is menu-driven and there is context-sensitive help.

While Eye-Q has its own printer driver, Ultraprint allows you 22 distinct styles/sizes of printer output. The reasoning is that the scale of gradation suitable for pictures is probably unsuitable for text or line drawings.

## PC CONQUEROR
## SOLUTION

PC Conqueror makes your QL into a PC-compatible machine,automatically. It does this by software means only, so there are no screws to undo or wires to fiddle with. Your QL stays a QL too.

Why, might you ask, should you wish to make your QL into a PC-compatible? The reason is simple: you may wish to run the same programs at home as you do at work. Alternatively, you may wish to tap into the vast storehouse of PC software of every type and description you could imagine.

Using PC Conqueror could not be easier. Just boot up your machine with the PC Conqueror disk in floppy 1 and within 10 seconds your QL will be transformed into a PC that is just waiting to be switched on. From this point on you will do exactly the same as you would if you were running a 'real' PC – this means putting a DOS disk (any version) into one of your drives and pressing a key. If you do not already have legal access to a copy of DOS, we can provide you with one at reasonable cost (see our price list).

PC Conqueror runs as fast as it is possible for a PC emulator to run: we have used all our skills to make it work quickly. Of course, you can make the emulation must faster by using Gold Card and Lightning SE. With this combination, you should get speed noticeably better than that of a PC XT...

PC Conqueror allows you to fine-tune the operating environment of the PC in order to improve performance. If you get a hard disk or other high capacity floppy system, you can utilise part or all of it as a PC hard disk.

PC Conqueror occupies under 80K and leaves 667K free for DOS when run on a Trump Card. This is more than you will get on a 'real' PC.

Solution does what Conqueror does but is about half as fast and is not quite as compatible.

## SPELLCHECKER
## MEGA DICTIONARY

Spellchecker is what makes Perfection into Perfection Plus. We have made it available as a separate item for two reasons: (a) to allow Perfection owners to add it later (b) to allow users of other word processors to benefit from the very best in spellchecking technology.

Spellchecker is supplied complete with three dictionaries of differing sizes as well as a system for building, reviewing and maintaining user dictionaries.

Spellchecker's ultimate accessory is the Mega Dictionary, which gives the user a vocabulary of over 350,000 words!

## 3D PRECISION CAD SYSTEM

This program allows you to manipulate shapes and figures in 2D and 3D at a speed that will leave you breathless. Irrespective of whether your interest is in CAD, in animation or in just having fun, this program should not be missed. You can output to plotters directly from it, or alternatively create graphics screens to be manipulated and output by Eye-Q, Ultraprint or Professional Publisher.

## SUPER SPRITE GENERATOR

SSG moves things about the screen very fast and very smoothly, without flicker. Sprites can have up to 16 frames.

# TROUBLESHOOTER

**Software updates, mysterious founts and clock comments from Bryan Davies.**

The version of *Lightning Special Edition* which is currently being shipped has some detail improvements over earlier versions. The changes are not likely to be noticed on basic (8-bit) QLs, but will be evident on systems with the Gold Card, Thor XVIs, and Ataris with the QL emulator (16- or 32-bit systems).

Transfer Utility Special Edition has had several changes made to it over the past months and now handles hard disk sub-directories and long file names. A comprehensive set of sorting and naming functions are provided. These latter functions make the program interesting to use even if you don't really need it.

Jochen Merz has produced a version of the Atari QL emulator to run on the Mega STE model, when plugged-into the VME bus. It will work on the other ST models also, with the aid of a converter. There is 256 KB of video memory and a 1000 x 1000-pixels screen can be handled - with any monitor, apparently. This allows the display of two pages side-by-side when using *text87* (which is said to be fully functional with the new emulator). This version of the emulator should work with the Atari TT (68030 chip, 32 MHz, 32-bit) too. My knowledge of the Atari models is sketchy, so interested readers should ask Jochen for full details. Performance is clearly going to be very brisk, and the display mode looks to be a big improvement on what we have available to us on the QL.

Software87 have a close working relationship with Jochen Merz and will be selling some of his software in future (QDesign, for example). They will also be selling QPac.

## Founts demo

There was only one microdrive cartridge in my QL "in" pile, and past experience leads me to expect trouble on the odd occasions a cartridge is inserted into my system, but this one had been waiting in line for attention for a year. The note attached to it said it was a scaled-founts demonstration, lasting one hour, and that sounded interesting. What was on the cartridge was indeed very interesting but, with the absence of the Microdrive Exchange, there is no obvious way to share it with readers. It was a demonstration of founts in various sizes, but scaled, as the note said. That is, the character shapes were displayed at gradually-increasing sizes, something I can't recollect seeing on the QL before. In addition, there was a demonstration of characters displayed on the screen in simulated 3D fashion; an example is shown in the illustration here, of a globe with *QL World* and other text written on its surface.

Using a Gold Card, with the files transferred to disk, the whole demonstration took appreciably less time than the hour quoted. Thanks to Stephen Poole of Le Bourg in France for a very well thought-out and constructed demonstration.

Mention of putting text in columns using Text87 Plus-4 sparked off a reminder that it has been possible to create text columns in Digital Precision's *The Editor* for several years. The process is not exactly intuitive, but it is fairly straightforward once you have tried it and grasped the principle. For anyone who still hasn't tried it, here is a typical command file for creating two columns:

```
T BTK SR32 JR
RP PR
T
RP (SM; 66N; BS; 65N;
31CR; BE; CM; 36CR; BM;
66N; EC..L.; N; 65D)
```

Before going into the construction of this command file, mention needs to be made of a "translation" problem - the "L" shown near the end of the last line of the file should have a bar above it, being the character created in The Editor by keying Ctrl-L. Unfortunately, I can't think of a way of putting it here that will stop it getting lost in the print process! The screen dump given here from The Editor shows some text after it has been "treated" with the command file. For assistance, the text for the commands is shown at the bottom (the L-bar character appears there); all that needs to be done to create the command file is mark the block containing the command text and save it as a _CMD file (then use Block Delete to get rid of it from the text). Type your basic text, correct any errors and do any other editing that is necessary, then run the command file by keying:

```
F3
RC.FLP1_COMMAND_CMD
ENTER
```

using whatever drive and file name is appropriate. The command line at the bottom of the Editor window shows this command inserted ready for the Enter key to be pressed. If the commands aren't basically clear to you, a few minutes with the instruction manual are indicated. The first line puts the cursor at the top, sets the block type to Column, then sets the width of the columns and the justification type. Left-plus-right justification is the most suitable form when columns are narrow. The second line causes the column width and justification

settings from the previous line to be put into effect. The third line merely puts the cursor back at the top, ready for the real action to start.

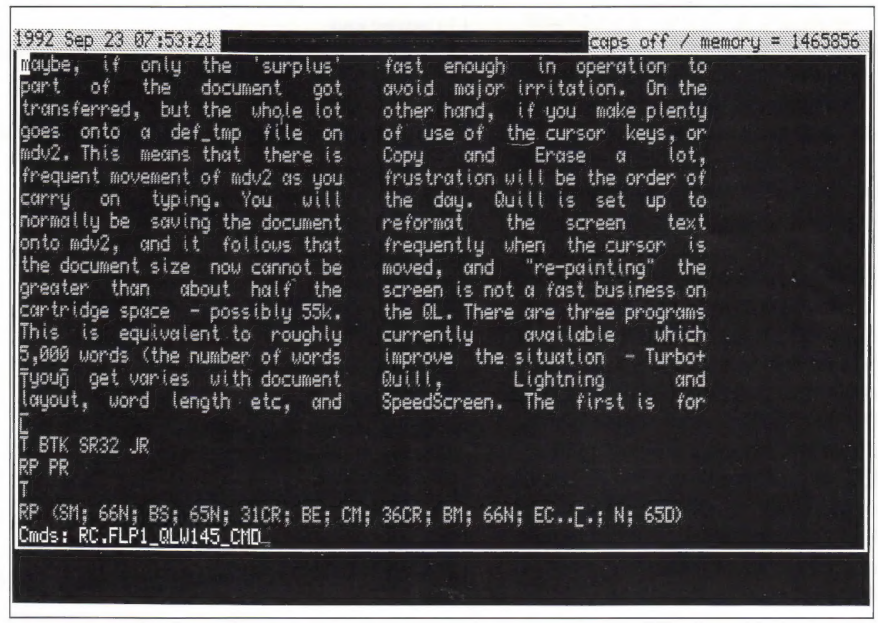
The final line is a set of repeated commands; the 66 and 65 figures are derived from an assumed page length of 66 lines, and you should insert your own values if a different length is used. The 36 is the column width of 32, plus a gap between columns of 4; again, set whatever is appropriate to your text format. Coming to the L-bar character, the command here is to insert that character - which is the code for end of page - after each block of 66 double-column lines has been created.

## Disk stuff

It could be argued that floppy disks are, by now, becoming close to obsolete. Developments in other forms of mass storage are trying to put the floppy in the shade, but there can be only a very small proportion of micros which sell without at least one floppy drive fitted, and that situation doesn't appear set to change in the near future. Removable hard disk drives are expensive and not common. Compact disks are expensive, as are the drives for them, and they suffer from a variety of not-quite-compatible formats; they are anyway not geared to the transmission of small, single programs, being aimed more at large databases. There is a definite trend towards supplying software for program developers on CD, though.

The wonders of credit-card storage are with us, but at a price few will want to pay. A special interface port (PCMCIA) is required, of a type that is only beginning to appear, and only on a few portable computers. Despite the (limited) availability for some years of floppy drives giving up to 20-30 MB per disk, even the straightforward 2.88 MB type is only now starting to appear in the mass market, and it may be too late for it to become a standard in the way that its predecessors have. The 4 MB floppy, which was said to be following the 2.88 one closely, doesn't seem to rate a mention in the popular computer press these days.

On my QL, the absence of the hard disk drive hasn't really worried me, and the ten ED disks which Miracle supply with dual ED drives have so far proved quite enough for my needs. If you haven't used the Miracle 3.2 MB drives on the QL, you can't fully appreciate what a difference the extra capacity makes to your daily operations.




## Clock quirk

Try this one, courtesy of Freddy Vachha of Digital Precision. Type-in the following SuperBasic lines, then RUN the routine, and see if you can make sense of the output on the screen:

```
100 FOR Z=1970 TO 2070
STEP 10
110 SDATE Z,1,1,1,1,1
120 PRINT
DATE$,DATE$(DATE),DATE
130 END FOR Z
```

Presumably, the QL clock was not designed to expect dates beyond about 2029, although that doesn't appear to be stated in the *QL User Guide*. Does everyone know about the odd behaviour for later dates? (Not that most of us will be too interested in them!) If that one doesn't appeal to you, try figuring out why the two PAUSE commands had to be inserted into the following line for JM QLs only - to get Perfection's Backup routine to work properly on that version:

```
19 PRINT"BACKUP"
:OPEN_NEW#4,dev2$&"BACK
UP":PAUSE 33:PRINT#4,'1
device$="'; dev3$;'"':PAUSE
33:LIST#4,2 TO 24: CLOSE#4:
DELETE dev2$&"BOOT"
```

This comment is directed primarily at programmers and suppliers of software, but it might ring a bell with users too. The QL has a clock function built into it but, unfortunately, it works only so long as power is supplied to the computer. A battery-maintained clock, as provided on the Gold Card, increases the utility of the function considerably. It is not really difficult to insert the correct date and time whenever a basic QL is powered up, though. To assist in doing this, a simple routine can be incorporated in a boot file, to prompt for the date and time to be keyed in.

What this is working around to is the failure of some programmers and suppliers to date-stamp the software they send in for review. A reviewer often receives several versions of the same program within a short period of time, and it is unlikely the disks will be the only things in the working area, so that the problem of deciding which is the latest version may arise. The date stamps on several copies of a program received for review recently were all in the 1960s. This wouldn't have mattered if the main program files had a version number at the front of them, but they didn't have one; the updates document on each of the disks referred to the latest version number, so there was no help there either. Programming should be an orderly business - please extend the orderliness to the documentation, programmers.

A time-setting SuperBasic boot routine is illustrated. It is based on one supplied originally by Eidersoft, with additional display procedures from Mike Lloyd's collection. There have to be shorter ways of writing such a routine than this, but it serves to show what is required. A basic assumption is that the entire date and time are set wrong if the year is found to be wrong, and the "right" year is 1992; you have to change two entries of "1992" in subsequent years. Don't assume that having a battery-backed clock ensures the date and time will always be correct; I've found them incorrect several times on my own system, sometimes as a result of interference from software that was being tested but other times for no apparent reason.

## Not Fair?

What has happened to the All Formats computer fairs? Not so long ago, they were occurring at London venues almost every month, but now they are few and far between and are being held at locations which may not be convenient for many of the people who used to visit the ones held in the Horticultural Hall. Are suppliers finding the faithful QL users still coming to the London fairs, or are the new locations putting people off? *(QL people - especially people coming from the Continent - don't find either Sandown Park or the Novotel, Hammersmith, anything like as*

*easy to get to as the Horticultural Halls. Although Hammersmith is fairly central and right on a major Underground line, it is nowhere like as central as Westminster. Ed.)*

AR Kempton got answers to his problem with the Taxan printer via letters in Open Channel, but a further comment is worth making. As the printer may be accessed by a "non-program" - eg the SuperBasic command line - when it has finished its job for Quill, it is desirable to do a complete reset at the end of each job, rather than at the start of each job. Putting the ESC@ command into the Preamble of the Quill Printer_dat file will work fine, for Quill, but has no effect on anything that comes after, so I suggest putting the same command into the Postamble instead (or as well, to play safe). There is then no need to issue a CR command, as ESC@ does that anyway. Incidentally, it was good to hear a voice that isn't often evident to the QL public these days - that of Chas Dillon. While we don't see Chas' name as author of any big new QL programs these days, his advice is still widely sought during program development.

```
1 Update:Leave
47 DEFine PROCedure screen (chan,wide,high,x,y,col)
48 WINDOW#chan,wide,high,x+8,y+4:
PAPER#chan,0:CLS#chan
49
WINDOW#chan,wide,high,x,y:PAPER#chan,col:CLS#cha
n
50 BORDER#chan,1,0:BORDER#chan,4
51 END DEFine screen
52 DEFine PROCedure write (chan,txt$)
53 LOCal gap
54 gap = ' ' INSTR txt$
55 IF gap
56 PRINT#chan;!txt$(1 TO gap-1)!:write
chan,txt$(gap+1 TO)
57 ELSE
58 PRINT#chan;!txt$
59 END IF
60 END DEFine write
61 DEFine PROCedure update
62 screen 1,145,38,252,36,20
63 CSIZE 1,0:INK 0
64 buff=PEEK_L(163916):POKE_L
buff+12,PEEK_L(buff+8)
65 :
66 LET year$=DATE$
67 IF year$(1 TO 4)="1992" THEN GO TO 100:ELSE GO
TO 68
68 LET a=1992
69 CLS
70 write 1,"MONTH?":getnumber:CLS
71 IF validflag$="T"
72 LET b=num
73 IF b<1 OR b>12 THEN GO TO 70
74 ELSE
75 GO TO 70
76 END IF
77 CLS:write 1,"DAY?":getnumber:CLS
78 IF validflag$="T"
79 LET x=num
80 IF x>31 OR x<1 OR b=2 AND x>29 OR b=4 AND x>30
OR b=6 AND x>30 OR b=9 AND x>30 OR b=11 AND
x>30 THEN GO TO 77
81 ELSE
82 GO TO 77
```

```
83 END IF
84 CLS:write 1,"HOUR?":getnumber:CLS
85 IF validflag$="T"
86 LET D=num
87 IF D<0 OR D>23 THEN GO TO 84:END IF
88 ELSE
89 GO TO 84
90 END IF
91 CLS:write 1,"MINUTE?":getnumber:CLS
92 IF validflag$="T"
93 LET E=num
94 IF E>59 OR E<0 THEN GO TO 91
95 ELSE
96 GO TO 91
97 END IF
98 LET F=30
99 SDATE a,b,x,D,E,F:CLS
100 INK 0:PAPER 2:CLS:CSIZE 1,0:write
1,DATE$:PAUSE 50:CLS
102 END DEFine update
103 DEFine PROCedure getnumber
104 DIM validflag$(1),num$(2)
105 LET validflag$="F"
106 INPUT num$
107 IF num$="0"
108 LET num=0
109 LET validflag$="T"
110 ELSE
111 LET num="0"&num$
112 IF num=0
113 LET validflag$="F"
114 ELSE
115 LET validflag$="T"
116 END IF
117 END IF
118 END DEFine getnumber
119 DEFine PROCedure leave
120 screen 1,145,38,252,36,0:CLS:CSIZE
0,0:CSIZE#2,0,0:WINDOW 322,158,89,49:PAPER
0:CLS
121
CLS#0:CLS#2:WINDOW#0,512,20,0,236:PAPER#0,0:I
NK#0,4:WINDOW 256,206,256,0:PAPER 0:INK
2:WINDOW#2,256,206,0,0:PAPER#2,0:INK#2,7
123 END DEFine
```

# QL SCENE

## Quanta in Bristol Area

More details about the next meeting planned by the Bristol Quanta group on Sunday 29 November, from 10am, at the Walton Park Hotel, Clevedon, Avon, south of Bristol on the M5 motorway. The meeting runs from till 4pm as usual, with traders stalls in the main conference room and demonstrations in an adjoining rooms. There will be a second demo room available on the ground floor if needed. Please let the Group know in advance if you want a stand or a slot in either of the demo rooms.

The Group will be putting on their famous Bring and Buy sale again. There is a bar and refreshments. Lunch is available if pre-booked with the hotal.

Overnight accommodation can be arranged at the hotel or at nearby lodgings. Sample prices (bed and breakfast per head per night) are: Walton Park Hotel £40 single or double (0275 874253) Cracklewood £18 basic, £24 ensuite (0275 8774717) Moor Villas £14 single £26 twin (0275 872131). Book direct with the hotels. **Further information on all fronts from Mike Ashford on 0272 629981.**

## More PD Ware

SJPD Public Domain library has released its twenty-second disk of software. Disc 22 includes programs to print text files with accented characters, a text-column printer, address label database, a program-extension detector, a sector editor, a routine to add the QL under_score to files zipped on a PC, one game and other items. MicroEmacs 3.11 has arrived from Richard Kettlewell, and 60 disks of Continental software have arrived for sorting.

Like PD library owner Ron Dunnett (September *QL Scene*), SJPD proprietor Steven Johnson hopes that he will be able to find assistants to translate some of the Dutch, German, Spanish, French and Italian software into English.

SJPD also has a new archiving utility for the QL, QL-Zip. Expected in by the time this appears in print is Ergon Developments' ZX Spectrum emulator. SJPD's fifth catalogue issue appears on 1 December. The copy charge is £1 a disk, and the list is available from **SJPD, 36 Eldwick Street, Burnely, Lancs BB10 3DZ. Tel. 0282 51854.**

## Spain Calling

The Spanish QL users' group QL' per is now sending out its disk-based newsletter to other QL user groups around the world, hoping for a regular exchange of publications. They will also send and exchange with groups using MS-DOS formatting.

The QL' per disk is bimonthly, and material is generally free for use as long as the source and author, and the fact that it has been translated (if this is the case) are acknowledged.

A note is appended to say that some issues of disk 40 cannot be read, and 40 will be sent out again with 41.

QL' per can be contacted at **Acacias 44 (Monteclaro), E-28223 Pozuelo de Alarcón, Spain.** Comments on QL' per from anyone receving it regularly will be appreciated at QL World.

## Software87 Offer Exotic Wares

Software 87, renowned publishers of *text87,* is now also offering a range of software from Care Electronics, Jochen Merz Software and Cowo Electronic. *QSpread,* the new speadsheet that can handle up to 32,000 cells; the latest *QDesign II,* the page design and drawing program, that now also works with JM-rom QLs and Minerva's extended graphics, and *Qpac II,* the established file and system manager, are now on the list.

*QTop,* Cowo's alternative file manager, *QD* version 4.00 and *ArcED,* two program text editors, and *FiFi* the fast file finder (which can search all files on a disk and display those with the keyword selected by the user) are also available.

Sample prices on Software 87's list are: Qspread £54, QDesign II £54, QD £44, QPac II £39. A mouse with a serial cable is available at £49.

One advantage in buying software from Software 87's list is that it avoids the need to cover currency conversion charges and postage from exotic places like Germany, Switzerland and Watford.

Watford!?

Meanwhile, text87 Plus4 has now been updated to make use of all the resolutions available on the forthcoming QL emulator card for the Atari MegaSTE and TT models from Jochen Merz. Plus4 will be able to use the whole extended screen area to display up to eight documents simultaneously, with a maximum 95 text lines (up to two sheets of paper) in any one text window.

**Software 87, 33 Savernake Road, London NW3 2JU.**

Prices include postage. Leaflets about text87 available on request.

## DJC cares for Care

Dilwyn Jones Computing now has an agreement to sell most of Care Electronics' range of QL software (with the exception of Tony Tebby's Toolkit 2). The programs now on DJC's list are: Tony Tebby's QPAC1, £19.95, QPAC2, £39.95, QTYP £29.95; Steve Jones's Sidewinder Plus, £24.95, Locksmithe (mdv backup), £14.95; 4Matter, £23.50, Toolchest (mdv backup package, includes Locksmithe, £14.95, Toolshest (mdv doctor-writer utility) £14.95. All the programs except Locksmithe and Toolchest require an expanded memory QL. All programs except the mdv-specific items are available on disk only. More details to follow.

Meanwhile, the Jones family's latest production is Gareth Sion (Shaun to the Sassenachs), arrived 4 October at 7lb 12 oz, not available on mdv or disk, but possessed of a rapidely expanding memory (and lungs, no doubt). Congratulations to Dilwyn and Janet.

**DJC, 41 Bro Emrys, Tal-Y-Bont, Bangor, Gwynedd LL57 3YT.**

# OPEN CHANNEL

## QDesign

With reference to my *Qdesign* user report (*QL World* September 1992), on reading it I realised that I had accidentally left in a sentence saying that it was impossible to escape from a print dump. This is wrong. You must hold the Esc key down for 2-3 seconds before it suspends the print dump.

**AF Wilson**
**Larbert**

## Z88 rejection

I just received my May-June issue, and am glad to read that *QL World* is safely windsurfing back into orbit. I have a couple of small matters that need attending to. First, the US postal service is just as touchy as the UK postal service. Someone failed to include my zipcode - a Cardinal Sin to the US postal services. I do not like to see them too upset, as they have a nasty habit of losing your mail in the crevices of the post office.

Second, maybe one of your readers can shed some light on a slight problem that is troubling me.

The QZ program that transfers data from the Cambridge Computing Z88 has recently developed a neurological problem. It no longer wishes to play 'transfers' properly with my QL. Up until 1 May the QZ behaved correctly; now it no longer wishes to respond to the data transfer keys, the Receive or Batch from the menu, but aborts the program. It will Send data from the QL to the Z88.

The same problem manifests itself using the Sector original mdv cartridge, and also on my buddy's QL system. We both have beenusing the Gold Card with the minerva rom and ED drives. For months the QZ program worked perfectly. Then, out of the blue, it does not wish to play by the rules.

Again, welcome back on board.

**W H Woodward**
**Lake Butler**
**USA**

*Comment: One reputable QL expert hazards an educated guess that your serial port driver chip (the 1488) has blown.*

## Macs

I have had to start using an Apple Mac LC for a lot of work. I am happy with this machine, but miss the QL greatly. I do not have enough desk space to have both systems set up. I am aware that the 680X0-based Atari ST has a QL emulator available for it, and I wondered if there was a similar product for this 68020 Mac.

**D M Peberdy**
**Droitwich Spa**
**Worcs**

*Comment: Much to our vexation, no, there isn't.*

## More Abacus

Here is one solution to the problem of how to order Abacus in reverse. Suppose you have a job in the range A1:D10 and you want to sort rows 3 to 8 in column C in descending order.

First go outside the job, for example, to cell F3, and press 1/C3 <Enter>, which will place the inverted sum of C3 in cell F3. Then ECHO cell F3 over the range F4:F8. Command F3 E<cho> cell F3 <Enter> over range F4:F8 <Enter>. This will give you an F column with inverted values of the contents in the cells you want to order in reverse.

Now you can order your job in reverse by sorting column F, rows 3 to 8. Command F3 O<rder> F <Enter> from 3 <Enter> to 8>.

**Börje Johansson**
**Stockholm**
**Sweden**

*Comment: E E Stocker's request prompted the largest number of responses since Dane Kurth's letter about in the QL World (the replies to that letter have been too long to publish and very much on the lines of 'let's discuss this further'). There were three small diagrams with Borje's letter but regrettably they were too faint to print. I hope the process will be clear when put into practice.*

## Boomerang

Hi Friends!

Due to insufficient addressing, I got my metter to EE Stocker returned. Will you please forward it to him/her? I wish you every luck in the future with your magazine.

**Börje Johansson**
**Stockholm**
**Sweden**

*Comment: overseas readers please take note! Most people do not like their personal addresses to appear in a publicly-available magazine, so we do not print full addresses. We will forward letters if requested. Having said that. . . .*

## Africa

Here I am, miles away in Swaziland, off the beaten track, keeping myself amused with my two QLs and gradually learning more and more about them. But in my four years out here, I haven't found anyone else to communicate my discoveries to, in spite of having put notices in the local paper.

Perhaps QLers are so busy hacking, they don't have time to read the newspapers. As far as I am aware, no QL club exists in this area, but QL enthusiasts there are, I know, because I've seen QL World in newsagents both here in Swaziland, and also in Nelspruit, Johannesburg and

**Open channel is where you have the opportunity to voice your opinions in Sinclair QL World. Whether you want to ask for help with a technical problem, provide somebody with an answer, or just sound off about something which bothers you, write to:**
**Open Channel, The Blue Barn, Tew Lane, Wotton, Woodstock, OX7 1HA.**

obviously sells.

Now I want to get in touch with some of those people, so that we can form a mutual support group. I am appealing to you to publish this plea of mine in the pages of your excellent magazine, so that anyone who lives in the Southern African region will feel inspired to get in touch with me, so we can put some life into the Southern African QL scene! The address is Box 52, Mbabane, Swaziland. Tel. 42997. Many thanks.

**Paul M Williams**
**Mbabane**
**Swaziland**

## Archiving

Just to let you know how much I appreciate the Archive Answers programs, not in the least for their cute programming. The trouble is, I type them in, I enjoy them, and then I forget them until I need them, and then I can't find them and I don't even remember how to get them running. To solve this, I collected them all onto a separate disk, and wrote a small procedure which helps me find them and run them. That I used one of author Robin Stevenson's own procedures just proves how valuable they are. There are just three procedures, which I saved as ARANSWER, so the only thing I have to do is to run "ARANSWER" after Archive has been loaded.

```
proc ARANS
print: MODE
print at 1,10;"TOOLKIT PROCEDURES FOR ARCHIVE"
print at 2,10;"BY ROBIN STEVENSON"
print at 4,10;"0: totaling-procedures - may91"
print at 5,20;"command COPYSTRUC (to make temp file)"
print at 6,20;"commands TOTAL en AVERAGE"
print at 8,10;"1: print procedures may92"
print at 9,20;"commands TABLE en LIST"
print at 11,10;"2:form print procedures jul92"
print at 12,20;"commands with F keys"
print at 14,10;"3: fields (file structure) may92"
print at 16,10;"CHOOSE MENUE; 0, 1, 2 OF 3"
print
endproc

proc MENUE;OPT
if OPT=0: run object "TOTALS": endif
if OPT=1: run object "TABLE": endif
if OPT=2: run object "FORM": endif
if OPT=3: load FIELDS: endif
endproc

proc START
ARANS
endproc
```

To get a little more speed, I changed the subject programs in object files by saving them as "save object '"fielname"', and further provided them with a start procedure which opens again with a helpfile, showing a.o. the commands to be used, which then can be typed in on the command-line.

For TABLE the startprocedure reads:

```
proc start
cls
print at 3,10; "Insert data disc"
print at 4,10; "command:TABLE(LIST);"filename","logname",
print at 5,19; "option","extraline"
print at 6,10; "(multiline or single line output)"
print at 7,10; "ask directory for exact filename"
print at 8,10; "if no other open files enter "" for log$"
print at 9,10; "if Opt$="", output will be to printer"
print at 10,10; "TABLE needs any parameter if extraline required"
print at 11,19; "[" "] gives blank line, [""] gives no extra line"
print at 13,20; "NOW TYPE YOUR COMMAND"
endproc
```

Something like this can be made for Numberhandling (Nov 91) and Totalising (May 91), whereas in Direct Typing (Jul 92) the program contains its own start. Any new program running under Archive that suits your needs you can include in the first program.

So much for my own invention.

In return I would appreciate it if a correction could be published on the NewPage proc (June 92) where the last line is corrupted. I have some trouble getting a proper page break. If working Table, it asks for a new page after 2 to 6 lines, while when using List, the screen scrolls without making a break. I tried "let LINE = LINE+1: endif : endif ", but this makes no difference. For a moment I thought the bug was in PrintList, as this is a really intricate "if else, if else" construction, but I can't find anything wrong there, so in the end I gave up. I hoped someone else would have rung the bell already, but finding nothing about it in the September issue, please don't let me down.

**Albert van Rheenen**
**Amsterdam**
**Netherlands**

*Comment: Our apologies, Albert. We now know that Robin has been in touch with you, and we will try to get some of his comments from other readers' interest. Robin can be hard to pin down as he works away from home a lot. Unfortunately, as we received your letter on 16 September (a week after the October issue had gone to press ...) this is the earliest issue we have been able to mention it.*

## Postamble

In reply to A M Kempton's letter in the July QL World, he should type:

```
'cr,'ff
```

for the postamble code in the printer driver.

**Stewart Honeyball**
**Miracle Systems**

## Editor's notebook

Two trends are showing themselves in the QL market, both beneficial to QL users. One is for software and hardware producers to licence fellow-producers to carry their goods on a wider scale than before. This is especially useful for continental producers wanting to get their products to UK users, and vice-versa, because currency exchange charges on small transactions have always been a bugbear.

The other trend is for an increasing interest in Public Domain software. Diverse and devious though some of it is, the range of choice that it offers to users can hardly be without at least some small thing of interest to every user. There is also a greater trend towards shareware. There are some items of interest in this month's QL Scene on both subjects.

Apologies yet again for the alarmingly late issue (more detail in QL Scene). And more - we had to retrieve some mags from the Post Office because in our eagerness we double-printed a set of labels. If you get two copies - give one to a PC user. More haste, etc.

Fill in the survey on page XX - and send it to us. There's a free year's sub in it for whoever wins the draw. I know the Post Office gets the stamp-money - but console yourselves with the thought: somebody will be reading it, so you have *one* vote which may actually benefit you in these troublesome times.

# TRANS 24

**INFORMATION**
**Program: Trans24**
**Price: £10 (plus £1 post & packing outside UK)**
**Supplier: Dilwyn Jones Computing, 41 Bro Emrys, Tal-y-Bont, Bangor, Gwynedd LL57 3YT.**
**Tel. 0248 354023.**

**Trans24's purpose in life is to make 24-pin use of programs which only have 9-pin drivers. Bryan Davies tries it on his GQ 5000.**

This is a program which has me feeling that almost everything has been said about it once the price and supplier have been given. That is not meant to be uncomplimentary - the program is a simple one, it is cheap, and it works.

DJC specialises in the "no frills" end of the QL market, and it is good that this is so. There are quite a few QL users who don't want to spend over £50 for each program, and maybe don't want to tussle with complex programs and instruction manuals either. You can't have do-everything programs which need no instructions for £10, though. Trans24 has but a single purpose - to enable 24-pin dot-matrix printers to be used to advantage with programs which do not have 24-pin printer drivers.

## Examples

Examples quoted of programs which provide output for 9-pin printers, but not for 24-pin, are Easel, GraphiQL, PCB Designer 1, Page Designer 2, Desktop Publisher 2, Professional Publisher, QRam and Sidewinder. In fact, although earlier versions of these programs may not have 24-pin drivers, later ones of some of them do. The printers with which Trans24 is stated to have been tested are the Star LC-2410, all Epson LQ models, all Epson SQ and GQ models using LQ emulation, and the NEC 2200. For this review, an Epson GQ-5000laser printer was used, set to its LQ-2500 emulation mode when used with Trans24, and to its FX-80 emulation mode when comparison tests were made with 9-pin output from Professional Publisher and Easel. The LQ and FX are two of the 24- and 9-pin (respectively) standards you are most likely to meet emulated on other printers.

The printed instructions occupy seven A5 pages and are sufficient for most purposes. There isn't really much to be explained. Some notes are rather unclear, and a degree of understanding of QL operation is desirable, to be able to "read between the lines". One of the instructions is to set "TRA 0" if your QL has a JS or Minerva rom. While this might present no difficulty for a user who is also a programmer, the non-programming person is going to find such an instruction meaningless. Ignoring it did not appear to create any problem with the review on a JS QL.

Trans24 requires just a basic QL. It is primarily intended for use with floppy disk, although it is possible to use it with microdrives. The restriction is in the size of dumped files, which can be too large to fit on single cartridges. Multi-tasking is no problem, the F4 key being used to refresh the Trans24 screen. As graphics printing often takes several attempts before the desired result is obtained, it is convenient to be able to switch between your DTP or graphics program and Trans24, and not have to reset at frequent intervals.

## Creates

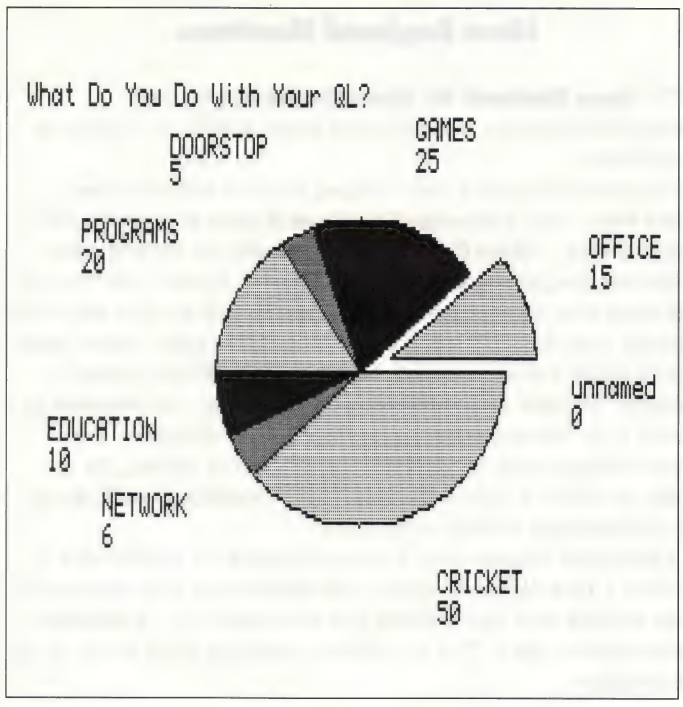
The program creates 24-pin printer output from the 9-pin output files of the originating program. It is therefore necessary to be able to send output to a disk file in the first place, and not all programs offer this facility. Easel is quoted as one program which won't allow output to a file, but there is a menu option in my version that does offer a screen dump to a file. This command produces a 32 KB file on floppy disk without trouble, but an attempt to print this file from Trans24 resulted in the all-too-familiar sight of sheets of paper streaming through the printer, with only a line or two of garbage at the top of each one. The instructions with Trans24 are written in somewhat quaint English in places, and the meaning in this area is not obvious at first. The intention seems to have been to say that Easel produces what would normally be called a screen dump - a pixel-by-pixel copy of the screen image - when the option to print to a file is chosen, and Trans24 is not designed to print from screen dump files. Unfortunately, the word "dump" is used in the instructions where printing to a file is referenced, which can cause confusion.

What is meant is that the GPRINT_PRT printer driver file has to be modified to make Easel print to a file in a way that will allow Trans24 to use that file. A small SuperBasic routine is provided to modify GPRINT_PRT suitably. The device name is changed from SER1H to what you choose, such as FLP1_E. Any print action from Easel subsequently will send the file to flp1 and give it the name E; you have to remember that a second print will overwrite the first, and so on. If you are in doubt about whether or not a file is a SBYTES command screen dump, or is a print to file, check the number of bytes in the file; a figure of 32768 bytes indicates a screen dump.

## Blacker

The different number of pins in the print head of a 24-pin printer create an obvious problem when the printer driver is for a 9-pin head; as the extra pins "don't exist", they can't be used. This results in black areas looking faint. Trans24 brings the other pins into use and so darkens these areas. Some text will look better because of this, although small text may look rather worse through character strokes being too wide.

The two illustrations are of prints made with Trans24, one from a picture provided with Professional Publisher (COMPUTERS_PIC) and the




# TRANS 24

other from a simple pie chart created with Easel. In both cases, the original pictures were printed to a file. The modified GPRINT_PRT file was used for the pie chart. The version of Professional Publisher used was the current one, which can print to a 24-pin printer, but the basic FX-80 driver was used here, purely to obtain a 9-pin disk file.

You have to bear in mind that the quality of the print depends very much on the program and driver from which the data originates and a 24-pin printer cannot miraculously produce photographic prints from "dotty" images. Trans24 appeared to alter print dimensions in some cases, but that was actually beneficial in the case of pie charts from Easel, as the shape of the pie was more circular. As I have found when making screen dumps for reviews, using 24-pin mode gives darker and more acceptable prints than 9-pin mode; a similar effect can be obtained by photocopying. While there is no real change in quality, the perception is of a better print. If you regularly make use of prints from a DTP or graphics program which has no 24-pin driver, Trans24 is an inexpensive way of gaining an improvement.

# QL SCENE

## Quanta in Ashstead

A new Quanta subgroup is planning to meet in Ashstead, Surrey, on the last Wednesday of each month. The first meeting took place on 28 October at St. Giles Hall, off Park Lane, Ashstead (Map ref. 192580, sheet 187. No excuses for not finding the spot!) from 7.30 to 10.30 pm.

For more information contact **Anthony Gordon, 13 Beales Road, Great Bookham, Surrey KT23 4NA. Tel. 0372 458180**, or leave a message on Tony Firshman's bulletin board on 071 706 2379.

## American Mag Into Second Year

American QL fanzine International QL Report: published by SeaCoast Services, entered its second year of publication this year. IQLR contains a mixed diet of news and product information, handy hints and tutorial articles, meeting reports and advertisements. [get the recent issue from Jim and fill in some contents information.]

The mag is around 40 pages of standard US-size paper (11 by 8.5 inches) with attractive parchment-look cover-pages, very readably-printed with about 450 words to the page - about the same quantity as Quanta, on average, but in a larger typeface. (For comparison, QL World has around 1400 words to a full page, in roughly the same typesize as Quanta). IQLR has a very similar 'feel' to Quanta; perhaps a little more 'spacy' with larger headlines and text in ragged-right rather than fully justified. That is where the resemblance ceases, however, as IQLR does not seem to have any user group or library facilities.

IQLR is published four to six times a year, with size determined by the amount of material. Subscription rates are US$14.95 per year in the US, US$16.95 in Canada, and US$28.00 for the rest of the world. Fees are accepted in US dollars, or "Pounds Sterling or DM (German Deutchmark) bank notes (currency) equivalent to the $US amount". (Bear in mind that it is risky sending currency notes through the post, and consider getting a $US money order if you wish to subscribe - but check that it is non-negotiable, or it will be no safer than cash.) Their subscription year runs from 1 May to 30 April annually; mid-term subscribers will be sent back issues for the current year.

**IQLR, Seacoast Services, 15 Kilburn Court, Newport, Rhode Island, USA.**

## New England Elections

The **New England QL User Group** has elected hardworking director pro-tem Ernie Wider to fulltime Director of its activities.

The elections were a little overdue, says NESQLUG News editor Peter Hale, because of pressure of work and a very club predicament - "When there are six posts and six people, only backroom bargaining can make it all work!" Actually, NESQLUG has more than six members, but obviously only six who are crazy enough to do the dirty work! Ernie himself has been unwell lately, but he made it to the election meeting. Medical opinion has it (whether officially or unofficially, we don't know. You will have to plumb your own experiences for the answer, no doubt!) that Ernie's obligation to serve a year as Director of NESQLUG is doing as much, if not more, to assist his continued health as all the ministrations of medical science.

A long-time Sinclair user, Ernie persuaded his department to get him a Thor. When he retired, his department were apparently keen to have their turn with the one-time Super-QL - but nobody knows how to use it. This looks like the time for Ernie to set up as a consultant ...

## Miracle on the Road

Miracle Systems will be exhibiting at the following fairs ad meetings in November:

14 November
All Formats Fair, Novotel, Hammersmith, London
28 November
Quanta Workshop, Manchester
29 November
Quanta Workshop, Walton Park Hotel, Clevedon, nr. Bristol
29 November
All Formats Computer Fair, City Hall, Candleriggs, Glasgow

Glasgow and Clevedon on the 29th? We've heard of Miracles, but surely even Steuart can't move that fast, can he? Unless (as has sonetimes been conjectured) there are two of him.

In fact, Miracle's Mike is going to Glagow; if he can't make it, they'll miss Glasgow this time, but Bristol is definite.

## Archive erratum

Half a line dropped out of the Archive proc NewPage on page 30 of the June QL World (as Robin and various readers have recently pointed out). The line should have read:

```
let line=1: endif : endif
```

## Late

We used to say that just about the only thing that couldn't be blamed for delaying a magazine was the weather. Now even the weather has its say. Our production team came back from an overseas windsurfing conference - September and October are popular for the large autumn waves - early in October and discovered that, in their absence, *QL World* had not only not been distributed, but had not cleared the printers. A certain amount of panic ensued. That is the answer to the much-asked question: "What happened to the October QL World?

We regret that the accumlation of events will also delay the production of the November QL World unless we can get it through the printers in record time. We apologize to readers for any incovenience and worry that this delay may have caused them. (To add insult to injury, Mark and Jim's week in chilly Brighton was almost totally wave-free.)

# A Question of Dots

## Part 4

**Geoff Wicks tells the story of The Logo That Never Clocked On**

In the Workers' Council we wanted to design a logo to go on all our agendas, minutes and publications to make them stand out from the other documents that circulated in the agency. I found myself more and more obsessed with one idea for this logo which I would never have dared to suggest in practice. On the *Professional Publisher* disk there is a clip art file 'Timepieces_pic' which contains among other things a picture of an alarm clock with a grinning face. It conjured up associations in my mind of the Workers' Council waking the company up, especially if the clock could be set to the time our council meetings began. As about half of the agency's staff had sleeping-in duties, most of whom were not in the best of humour first thing in the morning, it also appealed to my sense of fun.What has all this to do with desk top publishing? Well, this article is about trying to be creative in the design of your documents and adapting your fonts and clip art to your own needs. As an exercise I tried changing the time on the alarm clock from 2.38 to 9.30, and the result appears as **Figure one.** The clip art was loaded into *Eye-Q,* a copy of the clock was made and the image of this copy reversed. The hour hand of the reversed image was cut and pasted onto the original, giving a two-hour-handed clock. The original hour hand was erased and a new minute hand drawn in. This was less than a hour's work and the result was a tailor-made piece of clip art.

### Putting in depth

Some quite simple, but good effects, take much less time than this. For example, the use of shadow effects. This is shown in **Figure two.** Your message is written and a box drawn around it, this is cut and pasted to another place and the image inverted. The original is then cut and pasted onto the inverted image using a slight displacement and hey presto! you have a shadow. For extra effect a clip art star can be added, but this involves extra work tidying the image up. Another type of shadow is shown in **Figure 3.** Here the text is typed in using the normal font and then typed in again, but this time as an upside-down italic font in a stippled ink colour. The two lines of text are then carefully lined up by carefully scrolling until it is in the right place. Effects of depth are often used to add interest to a page, especially when business graphics are used. Unfortunately *Easel* has now become a little outdated because it only has two dimensional graphics, which look rather dull by modern standards. You could add interest by using symbols from clip art or symbols fonts. Alternatively you could try Mike Lloyd's 3-D bar graph program, or my *3-D pie* program *(QL World,* November 1990). It is also possible to use output from PC graphics programs, but more of that later.Both Figure three and my next effect shown in **Figure four** are based on suggestions in the Desk Top Publisher manual. One page of this manual illustrated various effects which could be obtained by using the program. If you can get hold of a copy of the manual it is worth looking at the page. Figure four illustrates the effect of printing a font on a background design. First you lay down the background (from font A in Professional Publisher) and print your text on it. For a good effect you can change the colour of the font from black to white and print out the text again over the original text, using a slight horizontal and vertical displacement. Both the QL design and the writing on the wall also give a good effect if the colours are inverted. I have not done this in the illustration as it does not give a good result using my printing technique.

### Printing techniques

For those readers who are interested, the illustrations for this series of articles have all been produced on Professional Publisher using reduced width and length printing and two passes of the printhead. My printer is a Brother 1409, which is a 9-pin dot matrix printer. It does illustrate the very high quality that can be obtained from Professional Publisher even using a simple printer.When using a background to letters or a stippled text it is important to bear in mind what printing technique you are going to use. With reduction printing the stipple becomes more dense and may become indistinguishable from plain black.If you look at **Figure five** you will see only five As, although in practice there are six. The top right hand one is indistinguishable from the background. There should also be white dots on the dark parts of the right hand As, but they cannot be seen. If you are using reduction printing you should use very pale backgrounds, sometimes so pale that they are only just visible on the screen. The use of stipples in letters can induce a panic in some users, because lurking in the depths

Fig. 1. The various stages of changing the time on a clip art clock - not an hour's work.


Fig. 2. A shadow effect gives depth - the addition of a clip art star adds to the impact.

of both *Desk Top Publisher* and *Professional Publisher* is some dreadful computer jargon. When you merge fonts or brushes you have to choose between OR, XOR and AND. You can of course try to find out what these terms mean by consulting the *QL User Guide,* or Jan Jones' *SuperBasic,* but if you are like me you will not be much wiser for doing so. However good the explanations of the terms may be, they do not really help you in understanding what the effect will be in your document. There is a fairly good attempt to do this in the Professional Publisher manual, by describing what happens when you merge a forward slash with a backslash. You can see the practical result in Figure five. As well as the effect of merging an A in the three different styles, try it first as a black letter on a light background and then as a white letter on a dark background.

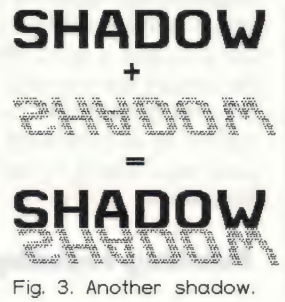


Fig. 3. Another shadow.


Fig. 4. Letters can be printed on a patterned background.

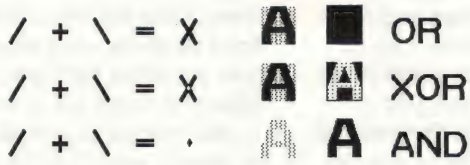


Fig. 5. Merging fonts in 3 different styles.

## Two colours

In practice there are only two colours in desktop publishing, black and white, so you can simplify the situation in a way that the computer purists would probably not approve by saying that in an OR merge black on black and black on white both produce black, in an XOR merge black on black produces white and black on white produces black whilst in an AND merge black on black produces black and black on white produces white. Try following this in Figure five.

Clip art can also be used to add effects to a page. There is quite a wide range of clip art available for the QL from a number of suppliers. In addition the Professional Publisher Toolbox contains a routine which permits you to use Sector Software clip art in Professional Publisher. Maybe it is heresy to write this in a QL magazine, and perhaps there are readers who will want to burn me at the stake for even thinking it but it is also possible to use PC clip art if you have Solution or PC Conqueror. The secret is to modify the boot program of Solution or Conqueror by replacing EXEC flp1_ by EXEC_W_flp1_. When the clip art is loaded press Ctrl-Shift-C to return to SuperBasic and save the screen by using the direct command SBYTES flp2_screen,131072,32768. (You should, of course, keep flp2_ free for Qdos, or alternatively save the screen to microdrive). You may find some slight colour distortion on your screen and you may have to recolour red or green to black or white. PC clip art often appears horizontally compressed on a QL and you may have to magnify it horizontally before using it in a document. If you do use PC clip art, however, be very careful about copyright. It may not be legal to use your firm's clip art in your private documents. Read the small print! Clip art can spoil your document rather than enhance it if it is incorrectly placed. It should draw the reader's attention towards the text and not away from it. In **Figure six** is an example of poorly placed clip art and two possible ways of correcting it. One of these ways involves electronic manipulation of the image, and if you wish to get the best out of your clip art you should not be afraid to do this.

## Recolouring

In this respect Professional Publisher is a very versatile program, since it contains routines for recolouring both mode4 and mode8 pictures in a monochrome texture; for inverting and reflecting the image or producing a symmetrical version of it; for slanting the image; for recolouring it; and for magnifying or reducing it. Alternatively the image can be manipulated in a similar way in one of the QL graphics programmes. One of the sad omissions from Professional Publisher is a routine for saving a screen so that it can be improved in a graphics program, although there is a program for doing this on the Professional Publisher Toolbox disk.

Magnifying and reducing images can be helpful when you need to fit a picture in a column, but can have disadvantages. When magnified some images will have jagged edges and when reduced some detail may be lost. In general magnification and reduction work better with solid images that with line drawings. If you have a copy of my last article look again at the steam engine illustration. The one in which the image size was changed was magnified once horizontally and then reduced twice horizontally to get it into a better proportion.There was

This picture is taken from the *Professional Publisher Toolbox*. If you put it in the wrong place this woman could be looking out of your document and distract the reader's attention from your text. You can either swop the columns or keep the columns and reverse the image electronically.

This picture is taken from the *Professional Publisher Toolbox*. If you put it in the wrong place this woman could be looking out of your document and distract the reader's attention from your text. You can either swop the columns or keep the columns and reverse the image electronically.

This picture is taken from the *Professional Publisher Toolbox*. If you put it in the wrong place this woman could be looking out of your document and distract the reader's attention from your text. You can either swop the columns or keep the columns and reverse the image electronically.

Fig. 6. The placing of a picture is very important. It should not distract from the text.

little loss in quality. Slanting an image helps to give depth or in producing a shadow effect. It is quite impressive to see the routine in action. Symmetry allows half the screen to be reflected vertically or horizontally. Do it once vertically and once horizontally and you can obtain a kaleidoscope effect. In **Figure seven** this is done using a clip art quillpen and modern pen. In my final example I am afraid I am going to bore you with politics. The reason I am doing this is because I have suitable clip art available. Suppose I have written an article on the political parties in Britain and wish to illustrate it by referring to the result of the 1987 General Election. I have two suitable pieces of clip art. One is a map of mainland Britain showing where support for the main parties is and the other is a three-dimensional bar graph prepared on a PC business graphics program. Both are mode 8 screens in 5 colours.

Fig. 7. A clip art kaleidoscope effect.

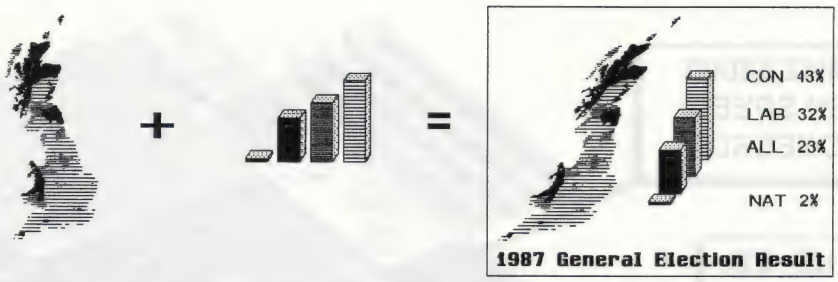


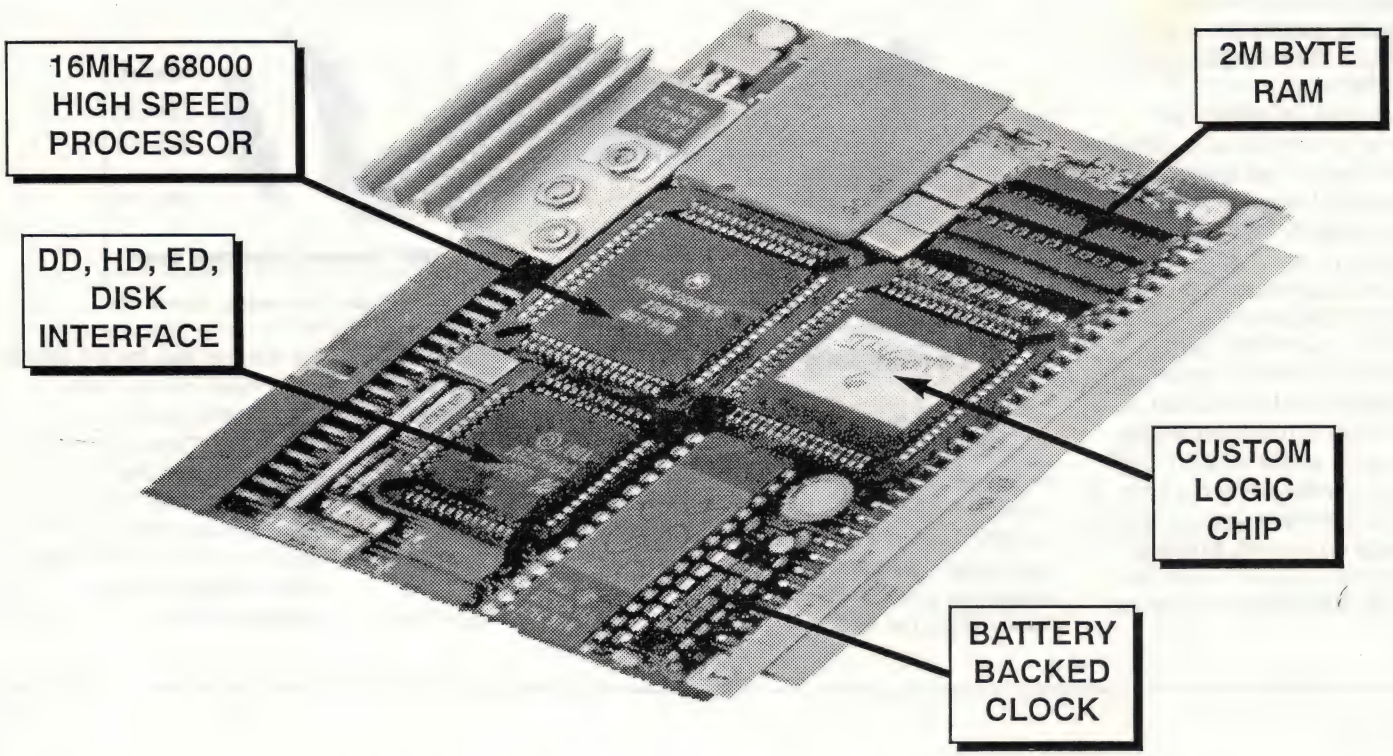

Fig. 8. Combining two separate clip art pictures into a more interesting illustration.

## Combining

Firstly both screens are recoloured using the mode 8 texture routine in Professional Publisher. Each could be used on their own but they would lack impact. **Figure eight** shows the effect of combining them. Firstly the map is slanted to give an impression of depth and the bar graphs are added using clip and paste methods so that they overlap one another. The illustration is put in a box and captions are added. It took only a few minutes to do and the result is a more professional-looking picture.

For me this was the fun article and I hope you will enjoy experimenting with my suggestions just as much as I did when trying them out. In the next article it is back to more serious matters and a little bit more humdrum work when we take a look at designing fonts.

# MIRACLE




# QL GOLD CARD

## £225 inc. (£200 export)

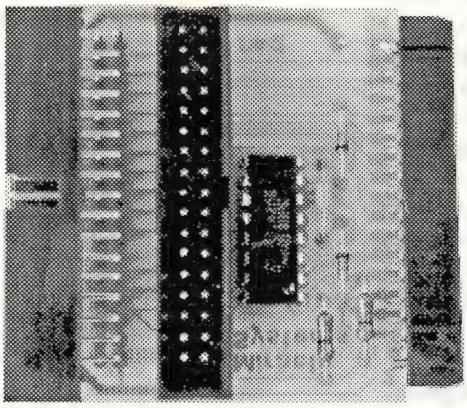
This is the expansion that has been revolutionising the QL. It is very easy to fit - it simply plugs into the expansion port at the left hand of the QL - and once fitted it will instantly increase the execution speed of the QL by about 4 times due to the presence of a 16MHz 68000 on board. There is 2M of fast 16 bit RAM of which QDOS sees a contiguous 1920K. The remainder is used for shadowing the QL's ROM and display memory and for the GOLD CARD's own code.

There is a disk interface which can access 3 mechanisms (4 with the DISK ADAPTER) of 3 different densities, DD (double density, 720K), HD (high density, 1.44M) and ED (extra high density, 3.2M) in any mix. The disk interface connector is the same type that was fitted to the TRUMP CARD so most QL compatible disk drives can be used. Please note that DD drives still give a capacity of 720K per diskette. Our DUAL ED DISK DRIVE allows the GOLD CARD to access DD, HD and ED diskettes.

Another feature is the battery backed clock. When the QL is switched on the contents of the clock are copied into the QL's clock so that the time and date are correct. The firmware in the ROM gives the GOLD CARD all the functionality of the TRUMP CARD like TOOLKIT II and there is a sub-directory system for floppy and RAM disks.

Physically the GOLD CARD is about half the size of the TRUMP CARD and so fits almost all within the QL. Its current consumption is well under the allowable maximum so no special power supply is required. The GOLD CARD comes with a 14 day money back guarantee and a 2 year warranty.

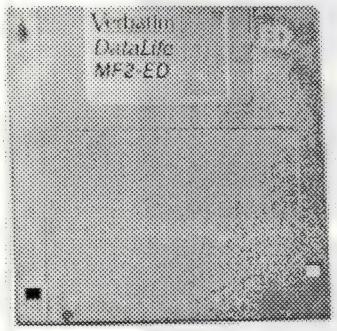
# SYSTEMS



## BOX OF 10 ED DISKETTES

**£30 inc. (£30 export)**

Ten 3.5" ED diskettes. Gives capacity of 3.2 MBytes when used with GOLD CARD and DUAL ED DISK DRIVE.




## DISK ADAPTER

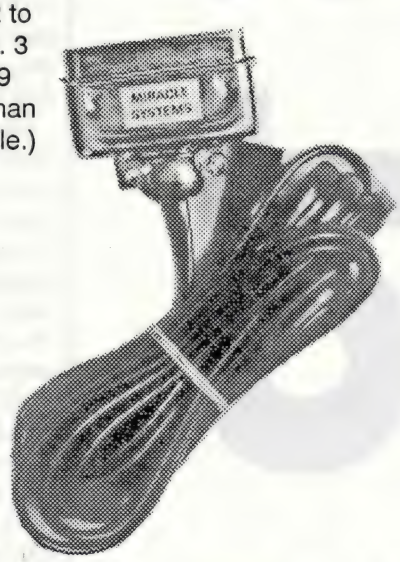
**£15 inc. (£15 export)**

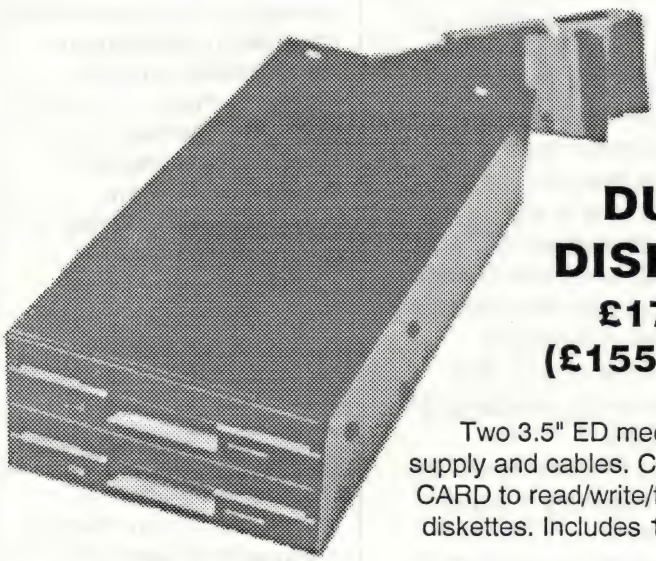
Plugs into TRUMP CARD or GOLD CARD to allow access to 2 dual disk drives (i.e. 4 mechanisms) as FLP1_, FLP2_, FLP3_, FLP4_.

## QL CENTRONICS

**£25 inc. (£25 export)**

Connects SER1/SER2 to Centronics parallel printer. 3 metre cable included. (D9 serial plug option for German and American QL available.)



## DUAL ED DISK DRIVE

**£175 inc. (£155 export)**

Two 3.5" ED mechanisms, power supply and cables. Connects with GOLD CARD to read/write/format DD/HD/ED diskettes. Includes 10 ED diskettes.

VISA MASTERCARD

## Tel: (0904) 423986

**To place an order by phone please have your credit card ready. For overseas customers we charge the prices shown in brackets.**

**To order by post, please fill in the form opposite or write to us quoting your credit card number and expiry date, or enclosing a cheque payable to MIRACLE SYSTEMS Ltd.**

**To: MIRACLE SYSTEMS Ltd, 25 Broughton Way, Osbaldwick, York, YO1 3BG, U.K. Tel: (0904) 423986**

Please send me _____

I enclose a cheque to the value of: £ _____

Or debit credit card ☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐

Expires ☐☐ / ☐☐

Name _____ Signature _____

Address _____

_____

_____

# QL SCENE

## Tidy SQLUG

You've heard of software, wetware and shareware. Now meet SQLUGware.

Something you step on in the garden? Not unless your post-person is unusually careless. SQLUGware is (and we quote) "a special kind of public domain software for the QL, designed specifically to increase awareness of the Scottish QL Users Group. Packages released under this banner will have been written by members of the Scottish QL Users' Group. They will be self-contained, fully-working programs with (hopefully) wide appeal."

In other words, as well as attracting attention to their own QL activities, SQLUG are supporting other QL users and support groups.

SQLUGware packages may be copied and passed on freely, provided that the package is intact and the documentation explaining SQLUGware is present.

User registration for this essentially shareware package can be on one of three levels: No registration, for light users, where the package is used as standard public domain software. Recommended for more serious users is either Level 1: fill in a form and return it to SQLUG with a stamped self-addressed envelope, and you will receive news of the next update when it is ready, after which you can obtain it (eventually) from your usual public domain sources OR send two 3.5 in DS/DD disks (one for your update, one as a donation to the group) complete with return postage and packing, and you will receive the next update. Level 2 (full) registration costs £5 UK (£7 elsewhere) payable to the Scottish QL Users Group (address below). For this you will receive upgrade disks as soon as they are available. One upgrade is guaranteed; up to three will be sent automatically. A printed version of the manual, plus upgrades, will be sent, and level 2 enquiries will be given priority.

Before registering, check that the package you are registering for is not more than two years old. If it is older, check with the group for current fees, terms of registration, etc.

SQLUG group members will receive information in the group's newsletter, and access to the group's library service.

They also very kindly recommend that users who can, support the QL by joining QL World and Quanta.

### The Program

Alan Pemberton's Disktidy has been made available as SQLUGware. Disktidy assists people with large disk collections by keeping a record of the directory of every disk in a collection. Disktidy allocates each new disk a number. When you have made significant alterations to the contents of a disk, run Disktidy to update the directory records. As well as the obvious usefulness of being able to scan all your disk directories, and search for filenames with Disktidy, the program also imposes a minimum necessary discipline on your disk collection by allocating disk numbers.

You will come to wonder how you every managed without it.

System requirements for Disktidy are twin disk drives OR a hard disk plus one floppy drive; SuperToolkit 2; ramdisks. The standard Trump Card or Gold Card sustems fulfil these requirements. If you have any queries (about the ramdisks, for example) no doubt SQLUG would be willing to assist. Free memory making the use of slave blocks possible gives faster operation.

### The Group

The Scottish QL Users' Group was set up in 1989 to provide a meeting-place for Scottish users. Meetings are held monthly 'in a reasonably central location, such as Stirling' to try out software, discuss developments and swap notes about problems. The newsletter reports on meetings and carries items of news and general interest. Membership is by subscription to the newsletter (currently £6 a year).

All enquires about the program and the group to Alan Pemberton, SQLUG, 65 Lingerwood Road, Newtongrange, Dalkeith, Midlothian EH22 4QQ.

## Hessler Handles Hermes

Sales of TF Services' Hermes intelligent peripheral controller in Germany are being handled by Albin Hessler Software, Im Zeilfeld 25, D-7447 Aichtal, Germany. The price is 79 Deutschmarks. Tony Firshman says: " Hermes is a good complement to Hessler's serial mouse driver, and version 2.04 or higher will allow the use of the Hermes split baud rates to drive a serial mouse at 1200 pbs at the same time as, say, a printer at 9600 bps." Albin Hessler has written some "more friendly" commands for setting the baud rate, which are now included with the UK release.

TFS are working on some new developments to Hermes, the split baud rates for the serial outlets being the most important. Combined with Hermes for serial input, this will give serial ports with completely independent baud rates.

TF Services, 12 Bouverie Place, London W2 1RB. Tel. 071 724 9053.

SINCLAIR QL WORLD

# THE NEW USER GUIDE

SECTION TWENTY

## KEYWORD INDEX

*This month in the Keyword Index, Mike Lloyd finally loops back to follow part 17 (INPUT$ - LRUN, August 1992) with MERGE filename to OJOB (job_id).*

### MERGE filename

PROGRAM FILE COMMAND

filename — A valid filename, e.g. "mdv1_myprogram"

The SuperBasic LOAD command automatically wipes out any existing program lines from the QL's memory before loading a new program. It can be useful to load extra program lines from a file without deleting what is already there, and this facility is provided by the MERGE command. It shares an identical syntax with LOAD. Care needs to be taken with line numbering to avoid existing lines being overwritten by new lines with the same line number (unless, of course, this is what is intended).
If the interpreter reads any non-SuperBasic lines while merging a file it marks them with the word "Mistake" and generates an error if the line is reached during program execution. If you are in the habit of writing program scripts using a word processor it is a good idea to MERGE them rather than LOAD them so that all errors of syntax are immediately marked.

MERGE can be quite useful to amend programs prior to running them. For instance, a program might have a number of user-defined procedures, only some of which are applicable to a particular configuration. Each procedure could be saved to its own file and merged with the main program only if it is required.

### x MOD y

MATHEMATICAL OPERATOR

x, y — Numeric values

MOD is not, strictly, a keyword: it is an operator belonging to the same family as the plus, minus and multiplication symbols. MOD is a form of division that gives the modulus, or remainder, when a value is divided by another. MOD answers the question "How many of these 37 oranges if I share out the largest equal number to my four friends?". Mathematically, the question can be phrased "PRINT 37 MOD 4". The QL guide indicates that MOD works only on integers, but it is more correct to say that it will work with any number, integer or real, and always returns a rounded integer value.

### MODE x

SCREEN RESOLUTION COMMAND

x — Numeric value

The MODE command establishes the screen display resolution, setting it to high or low with a corresponding change in the size of the colour palette. Whichever screen resolution is selected the QL allocates exactly 32K of memory to map the screen contents. To increase the resolution (the number of pixels on the screen) it has to reduce the amount of colour information held about each pixel,

which is why the high resolution mode has only four colours and no FLASH capability. Perhaps the most useful way of defining the mode is to use Mode4 to obtain the four-colour high resolution mode and Mode8 to obtain the eight-colour low resolution mode. However, any value between 0 and 7 indicates high resolution and any value between 8 and 15 indicates low resolution. To work out what display will result from any MODE value, use the formula "X MOD 16 - 8". If the result is negative the screen will be in high resolution mode, otherwise it will use low resolution.

Note that MODE does not set the QL's windows to their boot-up locations: this could be potentially disastrous while a program was running. To obtain the Mode4 or Mode8 window settings the WMON and WTV commands from *Super Toolkit 2* can be used.

## MOVE #chan

**distance**

GRAPHICS COMMAND

| | |
|---|---|
| #chan | A valid screen channel number |
| distance | A number of graphics units |

When the QL was being developed schools in particular were interested in the concept of turtle graphics pioneered by Samuel Papert. The idea was that children learnt basic programming skills from writing programs to control an imaginary "turtle" on the screen (which was, in reality, the graphics cursor). SuperBasic incorporates some turtle commands that can be used interchangeably with the more conventional drawing and trigonometrical keywords. Turtle commands refer to the graphics scale when determining distance: the default graphics scale divides the height of a window by 100. See the SCALE command for further information.

A turtle's orientation is established using the TURN command, its ability to leave a trail of ink pixels by the PENDOWN command. With a positive parameter, MOVE makes the turtle proceed forwards; a negative parameter makes it proceed backwards.

The difference between TURN and the analogous conventional command LINE TO is that it requires only one parameter to indicate the distance to travel whereas LINE TO needs two parameters to indicate a location. Which is better depends on the circumstances.

## MOVE_MEMORY source

**TO target, size**

## MOVE_MEMORY size

**source TO target**

**[Turbo Toolkit]**

MEMORY MANAGEMENT COMMAND

| | |
|---|---|
| source, target | Memory locations |
| size | Number of bytes |

It can be extremely advantageous on occasions to move chunks of memory around, although the operation has obvious dangers if the target location for a move is already occupied by something important to Qdos. Generally, *Turbo Toolkit*'s MOVE_MEMORY is used within reserved memory areas and the screen map. The command has a very wide range of uses from generating special effects on the screen to sorting database indexes at high speed.

The following program is a simple illustration of how the screen map can be manipulated directly to produce a special effect, in this case magnifying the top half of the screen display to twice its height. To understand the routine better you need to be aware that the QL begins its screen map at memory location 131072 and that each line on the screen occupies 128 bytes, no matter what screen resolution is in effect.

```
100 DEFine PROCedure expand
110 FOR x = 127 TO 1 STEP -1
120  source = 131072 * x * 128
130  target = 131072 * x * 256
140  MOVE_MEMORY 128, source TO target
150  MOVE_MEMORY 128, source TO target - 128
160 END FOR x
170 END DEFine expand
```

## MRUN filename

PROGRAM FILE COMMAND

| | |
|---|---|
| filename | A valid filename, e.g. "mdv1_overlay1" |

The SuperBasic LRUN command automatically wipes out any existing program lines from the QL's memory before loading and running a new program. It can be useful to load extra program lines from a file without deleting what is already there and without halting program execution, and this facility is provided by the MRUN command. It shares an identical syntax with LRUN. Care needs to be taken with line numbering to avoid existing lines being overwritten by new lines with the same line number (unless, of course, this is what is intended). When MRUN is used at the command line the QL begins execution from the first SuperBasic line, even if this was not one of those loaded by MRUN. If MRUN is used within a program, program execution continues from the command following the MRUN command.

MRUN could be used in a main program that calls a succession of merged files, all of which overwrite each other, in order to produce a sort of "overlay" effect. The practical difficulties are daunting, though, particularly with ensuring that each overlay had exactly the same number of lines with exactly the same range of line numbers. The effort may have been worthwhile before the widespread use of memory expansion cards and before *SuperCharge* and *Turbo* were available, but is now only applicable to very special circumstances.

## NET station

NETWORK COMMAND

station — An integer between 1 and 63

The QL was quite advanced in having a network protocol built into the standard model. (Contrast this with the completely different case of MS-DOS computers that have to have extra hardware added and system memory appropriated in order to communicate with each other.) Sadly, the Qdos network routines built into the QL's roms were crude and incomplete. In order to make any use whatever of networks it is a good idea to purchase Super Toolkit 2 for each computer on the net.

To give a QL a network presence, the NET command is used followed by a parameter representing a unique station identifier. If the network only connects two QLs they can share the same number, usually the default of 1.

Once a network is running it appears to the system as just another device. Channels can be attached to the device using a simple protocol. The following lines allow a document to be printed using a printer attached to a networked QL designated as a file server (see the FSERVE command). Note that the syntax is only available with *Super Toolkit 2* running on both computers.

```
100 NET 5: REMark use a unique identifier
110 OPEN #3, n2_ser1: REMark link channel to printer on file server (NET 2)
120 PRINT#3, flp1_mydoc
```

A more complete understanding of running networks is printed in the concepts section of the *New User Guide*.

## NEW

PROGRAM COMMAND

The NEW command simply wipes the QL's program memory area clean and shuts all channels other than 0, 1 and 2, the default screen channels. It does not, however, change their settings back to what they were on boot-up.

## NEXT ident

STRUCTURE CONTROL COMMAND

ident — A loop identifier, e.g. "X" in "FOR X = 1 TO 5" or "REPeat X"

NEXT has been embroiled in a contest with the END FOR command throughout the life of the QL. Programmers coming to SuperBasic from one of the other Basic dialects have traditionally used NEXT to end FOR...NEXT loops, but this habit is not encouraged on the QL even though the interpreter allows it. In SuperBasic, NEXT has special features that can be of great value. Firstly, it can indicate a premature end of an iteration of a loop, and secondly it can identify what has been called a loop epilogue.

Whenever it is not necessary to complete a full cycle of a loop, NEXT can be used along with an IF or SELECT ON structure to force a new cycle to begin. For instance, were we to examine each element of an array and only act on elements greater than 100 the following loop structure could be used:

```
100 FOR element = 1 TO DIMN(numbers)
110 IF numbers(element) <= 100 : NEXT element
```

```
120 REMark : The rest of the code.....
...
250 END FOR element
```

The same principle can be used with a REPEAT loop. Loop epilogues, however, are of value only with FOR..NEXT structures. The idea is that there might be some code that is only carried out if a FOR...NEXT sequence is completely exhausted; if the loop ends prematurely with an EXIT the code would not apply. SuperBasic uses an unconditional NEXT and an END FOR to allow this to happen. Using the example array above, we might want to exit the loop as soon as we find a value greater than 1000 and print a warning should there be no such values in the array. Here is the code:

```
100 FOR element = 1 TO DIMN(numbers)
110  REMark : Loop contents go here...
...
200  IF numbers(element) > 1000 : PRINT "1000+ value found" : EXIT element
210 NEXT element
220  REMark : The epilogue begins here.
230  PRINT "No 1000+ values found"
240 END FOR element
```

The inference from this variant of the FOR...NEXT structure is that END FOR is the correct way to terminate all FOR...NEXT loops. Purists might have a point: whenever NEXT ends a loop the interpreter will remain on the look out for a matching END FOR, occupying memory and slowing down interpretation for no good reason.

## NFS_USE device_id prefix, prefix prefix, prefix...

NETWORK DEVICE HANDLING COMMAND

| | |
|---|---|
| device_id | A valid device identifier such as "mdv" or "flp" |
| prefix | A valid prefix to a networked filename, such as "n2_flp1_" (Up to eight different prefixes can be assigned with one command) |

NFS_USE is closely related to FLP_USE. It allows device drivers to be redirected to point to other devices made available to a network. Just as "FLP_USE mdv" indicates that references to "mdv1_" and "mdv2_" should be taken to mean "flp1_" and "flp2_", the "NFS_USE mdv" command indicates which device represents mdv1_, mdv2_ and so on up to mdv8_. On a network of 2 QLs, the remote one being a file server, four microdrives and a distant disk drive could be referenced with the command:

```
100 NFS_USE mdv, n1_mdv1_, n1_mdv2_, n2_mdv1_, n2_mdv2_, n2_flp1_
```

This allows the local QL to refer to its own microdrives as mdv1_ and mdv2_ and the distant floppy disk as mdv5_.

## NXJOB(job_id top_job_id)

TASK CONTROL FUNCTION

| | |
|---|---|
| job_id | The name or identification number of a task |
| top_job_id | The identification number of the owning task |

All tasks are owned by other tasks except for SuperBasic, which always has an identification value of 0. NXJOB returns the identification value of the next job in the job tree headed by the given "top job". This is a very specialist function of value only to machine code programmers (and advanced *Turbo* users) and thus lies outside this Guide.

## OJOB(job_id)

TASK CONTROL FUNCTION

| | |
|---|---|
| job_id | The name or identification number of a task |

OJOB is a function to return the owner of a particular task. Thus NXJOB(this_job, OJOB(this_job)) will find the next job in the list to the one currently pointed to. It is unlikely that SuperBasic programmers will use this function.

# PSION SOLUTIONS

**Howard Clase makes sense of other machines' files on their way to Quill.**

After reading David Drysdale's comments in the April QL World, about the problems he had importing Z88 files into *Quill,* I thought it would be worthwhile looking into this topic in a little more detail. I had a SuperBasic routine for massaging MS-DOS files into importable shape, and have now expanded it so that it can extract the sense from more or less any file containing text in importable form. (It should even pick out the plain text messages from a machine code program - if there is any point to this.)

Let's start by examining the nature of a word processor file. Obviously the bulk of it will be the text you have typed in, but interspersed with this will be various control codes which tell the system when to change type styles, start new paragraphs, how to set the margins, etc. In a WYSIWYG word processor, like those available for the QL, these are used by the screen driver to display the text in the same form as it will eventually appear on paper.

## Printer driver

The purpose of the printer driver is to translate these codes into the appropriate ones for your printer at the printing stage. Some under-developed word processors on lesser machines only work properly at the printing stage. Just as different printers use different codes for the same thing, so do the different word processors' internal formats - which is why you cannot load a Word Perfect file directly into Quill, or probably any other QL word processor, even if you have already managed to transfer it onto a QL disk.

You can use your QL to examine a word processor file and work out what the internal codes are. If you have a Minerva rom chip you simply "COPY filename TO SCR" and all the control codes will appear as identifiable symbols - if you can remember what they all are! Alternatively, type in and run Listing two; this does a similar job, but presents each control code as its Ascii code, highlighted for emphasis and preceded by a space to separate consecutive codes. Since the codes corresponding to the QL second character set may well also be control codes, they are highlighted too. (You can easily modify the program so that it prints these as their Ascii values rather than as the characters. Just omit line 240.)

If you make a test _doc file with Quill, incorporating the typestyle changes at easily identifiable points you will find that, for example, Ascii 15 to 18 are used to toggle on and off bold, underline, low and high scripts respectively, that TAB is Ascii 09 (standard) and the end of paragraph marker is Ascii 0. Note particularly that there is no end of line (EOL) marker in the text, this is sorted out on the fly by the screen driver, and inserted by the printer driver at the last minute when you print.

## Psion import

The import facility in the Psion suite was designed to enable limited transfer of information between the four programs. They differ slightly: Quill is only able to deal with text in Ascii format, but it is possible to interchange numerical data among the other three as well. In fact the _exp files produced for importation into Quill are more or less straight Ascii files, oddly in MS_DOS rather than QL format. (This means they only contain the normal printable characters plus a few universally recognised control codes like LF (new line - CHR$(10)), and CR (carriage return - CHR$(13). MS_DOS uses CR,LF as EOL marker while the QL uses LF alone.) But import balks at files full of control codes, such as Quill _doc files.

Since Ascii files do contain an EOL marker at the end of each line, the import routine in Quill isn't quite sure what to do with them, so it asks a typically obscure question, "by line", - without even a "?"!. If you accept this by pressing <Enter> it makes each Ascii line a separate paragraph, which is the way to add a SuperBasic listing to a Quill _doc; but for ordinary text press <P> for "by paragraph", this causes the routine to ignore EOL markers unless it meets two together when it inserts a new paragraph. It is also clever enough to recognise CR and LF alone or paired in either order as EOL markers, so MS-DOS files and Acorn BBC files can be imported without any trouble. (I wish the reverse were true, and that Quill could convert each paragraph into a single Ascii line, empty of EOL markers except at the paragraph breaks, for export. One attempt by a long-time user resulted in 'single lines' - full of hidden EOLs, which popped out again one by one when we tried to reformat the file! But informed opinion so far is that Quill will not format a line long enough to contain a paragraph. - Ed.)

## Format file

I think this is the source of David Drysdale's problems since he was importing text "by line" and not by paragraph. The formatting commands, MARGIN and JUSTIFY work by paragraph, and since in his importation each line was a separate paragraph no wonder he found formatting tedious. In fact if you import by

```
5 REMark        Listing 2

10 nm$=           "Code_revealer"
15 REMark (pd) hjc  1992.06.24  ver 1.1
20 REMark ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^Main
25 Setup: Reveal_all: Finish: STOP
30 REMark ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
100 DEFine PROCedure                                                    Setup
105 CLS: CSIZE 0,0: INK 4: PAPER 0
110 INPUT"Device & filename:"\,n$
115 ch% = 3: OPEN_IN#ch%,n$:      REMark ch% unused
120 END DEFine : REMark ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
200 DEFine PROCedure                                          Reveal_all
205 LOCal k,k$,lp: CLS
210 REPeat lp
215   IF EOF(#ch%): EXIT lp
220   k$=INKEY$(#ch%,-1): k=CODE(k$)
225   SELect ON k
230     = 32 TO 127: PRINT k$;
235     = 10: PAPER 6: PRINT !k
240     = 127 TO 191: PAPER 6: PRINT k$;
245     = REMAINDER : PAPER 6: PRINT!k;
250 END SELect : PAPER 0: END REPeat lp
255 END DEFine : REMark ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
300 DEFine PROCedure                                            Finish
305 CLOSE #ch%
310 END DEFine : REMark ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
```

```
5 REMark          Listing 3

10 nm$=             "Filter_Ctrls"
15 REMark (pd) hjc 1992.06.22   ver 1.1
20 REMark ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^ Main
25 Setup: Filter_controls: Finish: STOP
30 REMark ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
100 DEFine PROCedure                            Setup
105 LOCal n$
110 CLS: INPUT"Device & file name:" \,n$
115 m$ = Psionise$(n$)
120 ch% = 3:                 REMark uses ch% & ch%+1
125 OPEN_NEW#ch%,m$: OPEN_IN#ch%+1,n$
130 PRINT\"Line counter ... "
135 END DEFine : REMark ^^^^^^^^^^^^^^^^^^^^^^^^^^
200 DEFine PROCedure                  Filter_controls
205 LOCal i%,k,k$,lp,n%,p%,s%: i%=0: n%=0: s%=0
210 REPeat lp
215   IF EOF(#ch%+1): EXIT lp
220   k$=INKEY$(#ch%+1,-1): k=CODE(k$):i%=i%+1
225   SELect ON k
230     =33 TO 127: PRINT#ch%,k$;: s%=0
235     =32 : IF NOT s%
240             PRINT#ch%,k$;: s%=1
245                REMark New line after 55+ chars
250             IF i%>55: Eol 0
255           END IF
260     =10,13: Eol 1
265 END SELect : END REPeat lp
270 END DEFine : REMark ^^^^^^^^^^^^^^^^^^^^^^^^^^
300 DEFine FuNction                   Psionise$(n$)
305 LOCal p%,m$
310 p%="_" INSTR(n$(6 TO LEN(n$)))
315 IF p%: m$=n$(1TO p%+4): ELSE m$=n$
320 IF LEN(m$)>13:m$=m$(1TO 13)
325 RETurn m$&"_psi"
330 END DEFine : REMark ^^^^^^^^^^^^^^^^^^^^^^^^^^
400 DEFine PROCedure                           Eol(f%)
405 IF f%: PRINT#ch%,k$;: ELSE PRINT#ch%,CHR$(10);
410    REMark Force a new paragraph every 55 lines
415 IF n%>=55: PRINT#ch%: n%=0
420 i%=0: s%=0: n%=n%+1
425 AT 3,17: CLS 4: PRINT n%
430 END DEFine : REMark ^^^^^^^^^^^^^^^^^^^^^^^^^^
500 DEFine PROCedure                            Finish
505 BEEP 5000,25
510 CLOSE#ch%: CLOSE#ch%+1
515 PRINT\"Control characters removed."\\
520 PRINT"New file name: "&m$
525 END DEFine : REMark ^^^^^^^^^^^^^^^^^^^^^^^^^^
```

paragraph and then go straight to the top of the imported section you will find that any changes to margins or justification will carry through right to the end of the file, or to the next point where you have already reset margins or justification if that comes first. The program "Filter_ctrls" given in listing 3 will filter out all control codes but the couple mentioned above from any file. I've tried it on all the types of file that I can think of - even a _doc file, and the results have always been readily importable into Quill. In addition to filtering out control codes (230 - 260) it removes multiple spaces (235 - 240), adds an EOL marker after the first word that takes the line length past 55 characters (250), and arbitrarily adds a new paragraph every 55 lines (415). This is because Quill has a maximum paragraph length (roughly a page of ordinary text), and refuses to import files that exceed this. There is a procedure (300) to produce a file name for the output which conforms to the Psion standard of a maximum of eight letters followed by an extension of not more than three characters. The extension doesn't have to be _exp, that's just the default, so I have used _psi. Change it to _exp if you would rather not type the extension in when importing.

While the main purpose of "Filter_ctrls" is to enable the importation of text files from "foreign" word processors, it could also be used with a corrupted quill _doc file, or even a def_tmp file in emergency recovery situations. There is no need to use it on a SuperBasic listing if you want to include one in some text, in fact the multiple space removal facility would remove all your structure indentations. The only requirement for importing a listing is that the name must conform to the Psion standard and you may have to rename it (or LOAD and reSAVE it under an appropriate name - "myprog_bas", where "myprog" doesn't exceed eight characters would be fine.) When you have imported your text a certain amount of tidying up will probably be necessary. The cursor will be at the end of the imported material so the first thing to do is to move it to the beginning (this will only be the "TOP" of the document if you started with a clean slate.)

## Margins, etc.

You should then find that the text is left justified and both the left and indent margins are set at the extreme left hand column. The right margin should be in its normal position in column 70 if you have imported the output file from "Filter_ctrls"; the position seems to depend to some extent on the line length of the imported file, which is why I built in the 55 - 60 character maximum; without it I found some files imported with the right margin in column 160! This is quite independent of the margins and justification of any text that is already present. If you then move the left and indent margins to columns 10 and 15 respectively and set right justification you will find that the whole of your imported text is now in the Quill default format. You will then have to remove the "false" new paragraphs, and if necessary insert "true" ones (if you have been recovering a lost Quill _doc you will find that all paragraph information has been lost, and there is some rubbish at the beginning and end of the passage.)

(In importing text for direct entry to a typesetting system, *QL World* has the opposite problem: for both style and technical reasons, they completely reformat imported text, and want Ascii text files free of all control codes apart from paragraph breaks and listing-indents.)

Since both "Code_revealer" and "Filter_controls" work through the file byte by byte they are relatively slow in operation. When using SELect ON loops it is a good idea to start with the most likely event and end with the least likely - reversing the order in Filter_ctrls added 30% to the time of operation. I have used integer variables wherever possible since they are faster on my Minerva rom, but if you have one of the Sinclair roms then it won't make any difference and you can omit all % signs. It is, of course, well worthwhile compiling these programs if you are likely to be using them often.

I have solved the problem of using more than one printer driver (printer_dat) file with the same Psion program by giving each a distinctive name, COPYing the driver to ram8_printer_dat as required and configuring Quill and *Abacus* to look for it there. (You can change drivers in mid session by using the backup command.) This means that I have half a dozen or so differently-named _dat files on each disk.

Some time ago, trying to work out how to wire up a QL to an Acorn BBC so that I could transfer files via their RS232 ports, I zapped ser1 on one of my QLs (or at any rate the handshaking part of it. That'll teach me to mess about with hardware!) So now I have one QL that prints from ser1 and another that uses ser2. Changing the serial port on a printer driver via install_bas is cumbersome, to say the least, so I wrote a small SuperBasic utility which sets the serial port on all _dat files on a device in one short operation. This depends upon the fact that the port is indicated by byte 16 of the printer_dat file, the one immediately following the 10 bytes set aside for the printer driver name. You can use "Code_revealer" to check on this. (Oddly, although install_bas will accept a name of up to 15 characters, only the first 10 appear in the code). Sensibly enough byte 16 is $01 for ser1 and $02 for ser2. It is important to use the correct port, because otherwise the Psion programs lock up and you have to reset the machine, losing anything in progress.

## Which device?

The program is given in listing four. You are first asked which device your _dat files are on; at line 140 the program calls a standard function I use to check that a valid device name is typed in (Get_dev$, 410 - 510) - this will add the

underscore if you leave it out (460) - and checks the device name against the list in p$ (480). If your device is not in the list, add it (420). While the PROCedure checks that the name is a valid Qdos device name there is no way it can check that you actually have the device hooked up, so you can generate a system error by putting in hdk1 when you don't have a hard disk! But it will trap typing errors of the type fpl2_, mdv22, etc.

There are of course only two serial ports so only "1" or "2" will pass the test in the loop 120 - 135; press the right key if you don't want a raspberry (130). This character, a$, is passed to the PROCedure Change_ports and coerced into an integer as n%; the other parameter, dev$ => d$, is the device name. The directory of the device is put into a temporary file on the device itself (210 - 215) - so you will need a bit of spare room on it. The names are then read one by one from the temporary file (230) and if they end in _dat (240) but are not "install_dat" (245) (that's the master file and we don't want to alter that!) the _dat file is OPENed and its name is printed onto the screen (250), the necessary change is carried out to byte 16 (255), and the file is closed (260).

(Since the pointer counts from zero not one it has to be set at 15 to change byte 16! For more info on the non-standard use of SCROLL to move the file pointer see Simon Goodwin's article in QL World February 1989.)

At the end of the list of files you EXIT the loop "change", CLOSE and DELETE the temporary file (270), and get a final message (405, 410).

## Conclusion

I hope that some of these routines will be of use as they stand, inspire you to adapt them, or write original SuperBasic utilities of your own. Remember, you cannot break your computer by typing words into it; but don't forget restrict your experimentation to backups.

```
5 REMark          Listing 4

10 nm$=         "Change_serial_port"
15 REMark (pd) hjc 1992.06.23 version 1.1
20 REMark ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^Main
25 Setup: Change_ports a$,dev$: Finish: STOP
30 REMark ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
100 DEFine PROCedure                          Setup
105 CLS
110 dev$=Get_dev$("Device containing _dat files")
115 PRINT"Serial port required (1 or 2)";
120 REPeat loop
125   a$=INKEY$(-1): IF a$ INSTR("12"):EXIT loop
130   BEEP 3000,25
135 END REPeat loop: PRINT\,a$\\
140 END DEFine : REMark ^^^^^^^^^^^^^^^^^^^^^^^^^^
200 DEFine PROCedure             Change_ports(n%,d$)
205 LOCal a$,change,L%,t$: t$=d$&"temp"
210 DELETE t$: OPEN_NEW#3,t$: DIR#3,d$
215 CLOSE#3: OPEN_IN#3,t$
220 REPeat change
225   IF EOF(#3): EXIT change
230   INPUT#3,a$: L%=LEN(a$)
235   IF L%>4
240     IF a$(L%-3 TO L%)=="_dat"
245       IF NOT a$=="install_dat"
250         OPEN#4,d$&a$: PRINT,a$
255         SCROLL#4,15,42: PRINT#4,CHR$(n%);
260         CLOSE#4
265   END IF : END IF : END IF
270 END REPeat change: CLOSE#3: DELETE t$
275 END DEFine : REMark ^^^^^^^^^^^^^^^^^^^^^^^^^^
300 DEFine FuNction                     Get_dev$(d$)
305 LOCal f$,lp,p$: p$="mdv\flp\fdk\hdk\ram"
310 REPeat lp
315   INPUT(d$&":-")\,f$: L%=LEN(f$)
320   IF L%=4 OR L%=5
325     IF LEN(f$)=4: f$=f$&"_": ELSE IF f$(5)="_"
330       IF f$(4)INSTR"12345678"
335         IF f$(1 TO 3) INSTR(p$): EXIT lp
340   END IF : END IF : END IF
345   BEEP 3000,25: PRINT
350 END REPeat lp: RETurn f$: END DEFine
355 END DEFine : REMark ^^^^^^^^^^^^^^^^^^^^^^^^^^
400 DEFine PROCedure                         Finish
405 PRINT\ "All _dat files on "&dev$;
410 PRINT " set to ser"&a$
415 END DEFine : REMark ^^^^^^^^^^^^^^^^^^^^^^^^^^
```

# DIY Toolkit

**Simon Goodwin explores the QL memory, with the latest DIY extension MAP in hand.**

This project uses SuperBasic, with existing *Toolkit 2, DIY Toolkit* and *Turbo Toolkit* extensions, to develop a new MAP command that can answer many interesting questions about the state of a busy QL. SuperBasic code equivalents and Minerva and *Qliberator* tweaks are also discussed, so you should be able to get the program running on any Qdos-compatible system, from the basic 128K model to the latest ST or Amiga emulator.

MAP_BAS, listed here, is derived from MEMORY_HEAP_BAS, one of the utility programs in DIY Toolkit Volume H. MEMORY_HEAP_BAS has generated much interest among QL enthusiasts, because it can answer the classic computer user's question - where has all my memory gone?

## Heaps

MAP_BAS demonstrates the general-purpose way that most QL memory is organised in 'heaps'. The code used to scan the Common Heap can just as easily be used to look through the Transient Program Area, occupied by QL tasks, as that uses an identical heap structure. It could easily be modified to scan the variable values in the SuperBasic heap, or user heap allocations made by the DIY Toolkit commands SET and ALTER.

**Figure one** shows the common heap map of a small QL system, with AH roms, 128K ram and Toolkit 2.07. The values may seem arbitrary at first, but you soon get to know the allocations in your own system, so that unexpected allocations stick out like a sore thumb.

The first heap entry corresponds to SuperBasic CON channel #0, used to display the start-up screen before F1 or F2, and later used for Basic command entry. The flag address points to the start of the Qdos channel table, where the ID for channel #0 is stored. The subsequent four areas are allocated by Toolkit 2 as it initialises itself.

## Defaults

Next comes the area used to store Toolkit defaults for PROG_USE, DATA_USE and SPL_USE, followed by the device linkage block shared by MDV1 and MDV2. This is initially in rom but copied to the heap by Toolkit 2, in order to add hooks for rename and truncate operations.

The Toolkit allocates the next area to two links in the polled task list, which also begins in rom. The standard routine ensures that the keyboard and serial inputs are read regularly. 288 bytes are then used to link the extra network devices, NET (replacing the rom device), N (used by the file server) and NFS_USE.

Once the toolkit has finished, Qdos sets up SuperBasic channels #1 and #2, using 256 bytes each. Subsequent blocks are allocated as tasks run. The next block is a 160 byte file channel definition. In this case it was used to send the map to a remote device over the network; the same space is used when the BOOT file is loaded after you press F1 or F2.

## Allocation

File-structured devices use three types of heap allocation. There is one device linkage block for each type of drive, such as RAM, MDV, FLP or WIN. When a drive is accessed a further 'physical definition block' is created to keep track of all files open on that drive. Individual channels have their own definition blocks, created when they are opened.

Each microdrive needs 1072 bytes to store its map of sector usage and table of pending operations. This physical definition for drive MDV1 comes next, followed by the smaller block used for details of the N1 network 'drive', and the physical definition for MDV2.

Space that has once been allocated but since released is marked << FREE >> and you might consider using the DIY Toolkit INVERSE command to emphasise it even more. Problems crop up when areas of free space are surrounded and split up by allocations around them. In the wrong place, even the smallest allocation of sixteen bytes can be enough to split a 400K free space into two chunks of less than 200K, restricting the maximum contiguous chunk of memory available for tasks, Basic, or program buffers.

## Contiguous

Many programs need large contiguous areas to run. You may be forced to reset the computer to get rid of an

Figure 1 - Diy Toolkit MAP "SCB" report for "AH" QL memory

System Variables & tables occupy a fixed 3584 bytes.

COMMON HEAP MEMORY

| Start | Length | Link | Flag | Owner |
|---|---|---|---|---|
| 167424 | 256 | 3242 | 166752 | SuperBASIC |
| 167680 | 128 | 1768 | Ø | SuperBASIC |
| 167808 | 8Ø | 1768 | Ø | SuperBASIC |
| 167888 | 8Ø | 1768 | Ø | SuperBASIC |
| 167968 | 288 | 1768 | Ø | SuperBASIC |
| 168256 | 256 | 3242 | 166756 | SuperBASIC |
| 168512 | 256 | 3242 | 166760 | SuperBASIC |
| 168768 | 160 | 168030 | 166764 | SuperBASIC |
| 168928 | 1072 | Ø | Ø | SuperBASIC |
| 170000 | 48 | Ø | Ø | SuperBASIC |
| 170048 | 1072 | Ø | Ø | SuperBASIC |
| 171120 | 912 | Ø | Ø | << FREE >> |

"AH" SuperBasic occupies 12288 bytes, with 77824 more available.

Figure 2 - QL World DIY Toolkit MAP report for "JSL1" QL tasks

TRANSIENT PROGRAM MEMORY

| Start | Length | Job Base | Flag | Owner | Name |
|---|---|---|---|---|---|
| 562176 | 480 | Ø | | << FREE >> | |
| 562656 | 10848 | 562760 | Ø | SuperBASIC | MAP |
| 573504 | 17600 | 573608 | Ø | QLTurboQuill+ | |
| 591104 | 19056 | 591208 | Ø | QLTurboQuill+ | |
| 610160 | 3536 | 610264 | Ø | QLTurboQuill+ | |
| 613696 | 62016 | 613800 | 357906 | SuperBASIC | QLTurboQuill+ |
| 675712 | 63104 | 675816 | Ø | SuperBASIC | Qlip72 |

allocation that splits your free space. This article explains how you can minimise that risk by understanding the way your QL uses its memory.

Chunks of free memory may appear in the transient program area when a task finishes, or in the common heap when a program releases memory or a channel is closed. Once allocated, an area cannot be moved as other parts of the system may contain pointers to it. MAP can locate and report on all allocations and free space in the task and common heaps.

You can also ask MAP to report on the memory used for SuperBasic, Resident extensions and System Variables. The current program does not look inside these areas, but extra code could easily be added, for instance to show the proportion used in each of the 13 dynamic areas within the SuperBasic space, or the utilisation of system variable areas.

## Task map

**Figure two** shows the Task Heap map for the Minerva QL which I am using to write this article. As is usually the case, it is not possible to make an observation without disturbing the thing you are looking at; the compiled MAP command itself appears as the first active entry in the task map.

Task memory is allocated from high addresses, towards lower ones, unlike the common heap, which starts at low addresses after the system variables and Slave Block tables. The heap allocation code works just as well, but new space is added when necessary at the lowest address, by moving SuperBasic down towards the common heap, perhaps squeezing out some slave blocks in the middle.

First my BOOT program loaded QLIP72, the latest version of the multi-tasking *Qlipboard*. Later, I used *Taskforce* to load *TurboQuill+*, which created three more spaces in the task area for its own data storage. Finally the MAP command itself creates a task, as it runs, using the *Task Commander* code from *QL World* October 1991.

```
Don't lose your memory!  QL World DIY Toolkit  MAP listing          page 1 of 3

100 REMark MAPS v 1.3 - TURBO version - uses #15
110 REMark QL World DIY Toolkit, PR Spink & SN Goodwin
120 REMark The next nine lines perform TURBO initialisation
130 :
140 DATA_AREA 1 :REMark 1K DATASPACE should be plenty
150 IF NOT COMPILED : OPEN #15,"SCR_480x210a0x0"
160 DIM t$(8) :REMark space for options
170 IF OPTION_CMD$=""
180   t$="*=CBT    " :REMark Default for patching
190 ELSE
200   t$=OPTION_CMD$
210 END IF
220 :
230 IF t$="?" OR "H" INSTR t$
240   PRINT #15; \"MAP options are:"
250   PRINT #15; \"A for ALL details"
260   PRINT #15;  "B for SuperBASIC"
270   PRINT #15;  "C for COMMON HEAP"
280   PRINT #15;  "R for RESPR space"
290   PRINT #15;  "S for SYSTEM usage"
300   PRINT #15;  "T for TASK memory"
310   PRINT #15; \"? for this message"
320   PRINT #15; \'e.g.  MAP #3,"BT"' : STOP
330 END IF
340 IF "A" INSTR t$ THEN t$="SCBTR":REMark ALL
350 IF "S" INSTR t$
360   PRINT #15;\"System Variables & tables occupy a fixed"!
370   PRINT #15;!PEEK_L(SYSBASE+4)-SYSBASE!"bytes."
380 END IF
390 IF "C" INSTR t$ : COMMON_HEAP
400 IF "B" INSTR t$
410   PRINT #15; \'"';VER$;'" SuperBasic occupies'!
420   PRINT #15;!PEEK_L(SYSBASE+20)-PEEK_L(SYSBASE+16)!"bytes, with"!
430   PRINT #15; !PEEK_L(SYSBASE+16)-PEEK_L(SYSBASE+12)!"more available."
440 END IF
450 IF "T" INSTR t$ : TASK_HEAP
460 IF "R" INSTR t$
470   PRINT #15; \"Resident procedures, functions and"!"devices occupy"!
480   PRINT #15;!PEEK_L(SYSBASE+32)-PEEK_L(SYSBASE+28)!"bytes."
490 END IF
495 CLOSE
500 :
510 DEFine PROCedure COMMON_HEAP
520   PRINT #15; \TO 16;"COMMON HEAP MEMORY"
530   PRINT #15; "Start" TO 10;"Length" TO 20;"Link";
540   PRINT #15; TO 30;"Flag" TO 40; "Owner"
550   HEAP SYSBASE+4 , SYSBASE+12
560 END DEFine COMMON_HEAP
570 :
580 DEFine PROCedure TASK_HEAP
590   PRINT #15; \TO 16;"TRANSIENT PROGRAM MEMORY"
600   PRINT #15; "Start" TO 10;"Length" TO 20;"Job base";
610   PRINT #15; TO 30;"Flag" TO 40; "Owner" TO 54; "Name"
620   HEAP SYSBASE+20 , SYSBASE+28
630 END DEFine TASK_HEAP
640 :
```

The information in this table is quite different from that shown by the usual JOBS or LIST_TASKS command. The owner of each area is shown by name, rather than by number. The memory used by each task is shown in bytes, with the start address, so you can PEEK and POKE inside the area if you're brave enough! The text buffer used by Spy, *The Editor* and *DevPac* is relatively easy to spot and PEEK. This can be a practical way to recover text from a crashed task, but the internal organisation of Psion programs is not for the faint-hearted.

## The Utilities

TURBO_MAP_BAS is listed here. It is an ideal SuperBasic extension, itself written in SuperBasic. You can compile it with Turbo or Qliberator, and convert the resultant task into a resident procedure with Task Commander, from DIY Toolkit Volume J.

The program illustrates how a channel and string parameter can be passed from SuperBasic to a Turbo task. Small changes allow the same for Qliberator, which uses #0 for the channel parameter, rather than #15, and the CMD$ function in place of Turbo's OPTION_CMD$.

TURBO_MAP_TASK, on the DIY Toolkit disk, is the same program, compiled to a little under 10K of stand-alone machine code. You can run this with EXEC, but you need Toolkit 2 or Turbo Toolkit to pass the channel parameter to it.

TURBO_MAP_CODE is the task file as treated by Task Commander. This accepts a parameter string from SuperBasic - try MAP "?" if unsure of the usage - and also lets you nominate a channel, defaulting to #1 to suit USE, as usual. To send the task map to the file FLP1_TMAP, type:

```
OPEN_NEW #3,"FLP1_TMAP" MAP #3,"T" CLOSE #3
```

Subsequently you can look at the file with MORE TMAP or any Ascii editor, or print it with COPY to an appropriate device.

## Memory analysis

The procedure COMMON_HEAP scans the common heap and shows all the memory areas inside. It prints five columns. First is the

address of the start of the area, then the size in bytes.

The next two entries are linkage values from the heap header. If the heap entry is a channel definition, LINK points to the linkage block of the relevant device driver. This entry can be used to distinguish between channel allocations owned by different devices. The first such entry normally corresponds to SuperBasic channel #0. The same LINK value is used by other CON and SCR display device definitions elsewhere on the heap.

FLAG is the address of the entry in the system task or channel table pointing at this heap definition. When the channel is closed or the task terminates, the byte at FLAG is set, so Qdos knows that the memory is no longer active and does not try to use the channel or schedule the task any more. SV.CHBAS (SYSBASE+120) and SV.CHTOP (SYSBASE+124) point to the channel table, and if FLAG fits that range you are almost certainly looking at a channel definition block.

You can derive the Task or Channel number from the value of FLAG; subtract the address of the first entry in the table from SV_JBBAS or SV_CHBAS, and divide the result by four, the number of bytes used by each pointer in the table. The result is the Task or Channel Number, which forms the channel ID in conjunction with the Tag word at offset 16 in the heap entry.



```
650 DEFine PROCedure HEAP(startp,nextp)
660 LOCal length,address,jobtag,jobnum,addset,start,free_ptr
670 REMark List heap blocks and indicate ownership
680 :
690 start = PEEK_L(startp)          : REMark first block
700 free_ptr = NEXT_FREE(startp)    : REMark first free block
710 :
720 REPeat scan
730   IF start >= PEEK_L(nextp): EXIT scan
740   length = PEEK_L(start)
750   address = PEEK_L(start+4)
760   jobtag = PEEK_W(start+8)
770   jobnum = PEEK_W(start+10)
780   addset = PEEK_L(start+12)
790 :
800   REMark is this block free
810   IF start = free_ptr
820     PRINT #15; start; TO 10; length; TO 20; address;  TO 40; "<< FREE >>"
830     free_ptr = NEXT_FREE(free_ptr) : REMark Find next free block
840   ELSE
850     PRINT #15; start; TO 10; length; TO 20; address; TO 30; addset; TO 40;
860     REMark Name of owner of area
870     PRINT #15; JOB_NAME$(jobnum,jobtag);
880     REMark This should only be true for the transient program area
890     IF PEEK_W(start+110) = 19195
900       REMark Name of job
910       PRINT #15; TO 54;PEEK$(start+114,PEEK_W(start+112));
920     END IF
930     PRINT #15
940   END IF
950   start = start + length : REMark Find start of next block
960 END REPeat scan
970 END DEFine HEAP
980 :
990 DEFine FuNction NEXT_FREE(start)
1000  REMark Find next free space in common heap
1010  REMark start = address of the start of a free block
1020  IF PEEK_L(start)= 0
1030    REMark End of linked list
1040    RETurn 0
1050  ELSE
1060    REMark return start of next free block
1070    RETurn start+PEEK_L(start+4)
1080  END IF
1090 END DEFine NEXT_FREE
1100 :
1110 DEFine FuNction JOB_NAME$(num,tag)
1120  IF  num = 0
1130    RETurn "SuperBASIC"
1140  ELSE
1150    sv_chbas = PEEK_L(SYSBASE+104)
1160    jb_ptr = sv_chbas+(num*4)
1170    REMark Is this job number in use ?
1180    IF PEEK(jb_ptr) <> 255
1190      jb_start = PEEK_L(jb_ptr)
1200      REMark Is the tag correct ?
1210      IF tag = PEEK_W(jb_start+16)
```

## LINK and FLAG

LINK and FLAG are important because they are used to control automatic allocations made by Qdos for tasks and channels. When heap space is allocated explicitly by programs their contents depend on the habits of that particular program.

You can normally spot memory allocated by ALCHP, ALLOCATION or RESERVE, as the same rom routine does the work in each case: MT.ALCHP, which puts characteristic values in LINK and FLAG. My AH-rom computer shows a link of 1768 for MT.ALCHP spaces, and 0 for memory allocated in supervisor mode by the device vector MM.ALCHP.

The last column is the name of the task that owns the memory - often SuperBasic, for system utilities - or << FREE >> if the memory is un-allocated. Often small free areas get left at the front of the heap, because subsequent requests ask for a larger space. Qdos and Argos merge consecutive free areas, sooner or later.

Directory device file definition blocks are 160 bytes long. CON channels with the full input buffer normally use 256 bytes of heap space. SCR channels use less, as they lack the input queue; on my QL, 112 bytes is normal, but this depends on the rom version. From JS onwards Sinclair added an extra long word at offset 100, at the end of the CON and SCR channel variables.

## Ramdisk

Areas of 544 bytes are used by the dynamic ram disk present in the QJump RAMPRT extension and the Thor rom. Every time a sector is needed, and not pre-formatted, the ram driver grabs 528 bytes - 512 for the sector data, and 16 for a list to tie the lumps together. The operating system adds another 16 bytes of common heap header, making 544 in all.

Pre-formatted ram disks are preferable, because they do not scatter allocations through the heap as they are used. Ram drives take 32 bytes each, plus 528 bytes for each sector, so a 100 sector (50K) ram disk needs 52832 bytes. It can store up to 50624 bytes, after allowing one sector for the directory and 64 bytes for a file header. These figures are correct for RAMPRT but may differ for other ramdisk drivers. The same file would use 54400 bytes, in 100 separate allocations, on a dynamic ram disk.

## FILL

Areas of 1040 bytes often correspond to the FILL buffers used by channels. These can be left in memory for ever if you accidentally CLOSE a channel after FILL 1 and before entering FILL 0. If you are sure that they are unused, you can remove them safely with Turbo Toolkit's DEALLOCATE procedure; you could use DISCARD if you POKE the magic word 'bufa' into the last long word of the heap header, usually 12 bytes on from the start address shown by MAP, which includes the 16 byte

header.

The QFLP disk system allocates 1616 bytes for each 720K drive that has been used, unless you stop using the drive and de-allocate the space with the Toolkit 2 command DEL_DEFB. Most of this space, from offset 80 onwards, is a copy of the medium header and disk map from the first block of the disk. The first section notes the number of files open (0-255), error status and details of up to 20 pending operations.

Unfortunately there is not enough space here to note all the contents of disk and microdrive system heap allocations. DIY Toolkit Volume H now contains a comprehensive list of heap variables used by QL and Thor directory devices. The file OFFSETS_DOC gives the labels, offsets, sizes and details of each part of the standard heap allocations for drives, file channels and devices.

## In use

Until you learn how your system treats the heap, MAP may be more useful for diagnosing problems than it is for solving them. It shows how your memory has become fragmented, in the hope that you can avoid causing the same problem in another session. It can also be very useful when debugging new programs that use the heap.

Memory is allocated whenever a drive is used for the first time, so it is a good idea to DIR all relevant drives, and expecially 'data drives' like MDV2, FLP2 or RAM1, before loading tasks, so that the drive details are stored before the task ones, where they are less likely to cause fragmentation.

Ramdisks should be formatted before use, and preferably before files are opened or tasks loaded. If an area is owned by a task it will be released when the task terminates, or you stop it with PURGE, REMOVE, RJOB or REMOVE_TASK.

Sometimes the Toolkit 2 command DEL_DEFB helps, by de-allocating the definition blocks, but that's a cautious command and makes many checks for open files and other possible problems. Often it decides to do nothing. If blocks are deleted they will re-appear in the first heap space big enough to hold them when the drive concerned is next used.

Another useful utility in DIY Toolkit Volume H is DEVLISTS_BAS, which scans through the system device lists, printing crucial addresses used by each device driver on your system. You may be surprised to see how many of these exist! The *Sinclair QDOS Companion,* by Andy Pennell, explains more about these values and lists.

DIR_DEVS shows entries for all the Directory devices, like FLP, RAM and MDV, while SER_DEVS scans for serial devices like SER, MEM, NET and CON. Notice how the addresses correspond to values shown by MAP and MEMORY_HEAP_BAS.

## Compatability

The program listed starts with a few lines aimed at the Turbo compiler. DATA_AREA 1 reserves the minimum 1K dataspace; alternatively you can do this from the compiler front panel or later with the DATASPACE_TASK utility. If running the program in SuperBasic, you need to open a screen channel with OPEN #15,"SCR_480x210a32x0" and set T$ to contain letters indicating the reports required. The default is "CBT", for Common heap, Basic and Tasks, but you can easily change this, even after compilation, by patching the text that starts "*=".

Most of the work is done by PROC HEAP, which scans a heap indicating the ownership of each block. NEXT_FREE is used to locate the next block in the linked list of free spaces.

The last three DEFinitions may be replaced with toolkit code. If you have Toolkit 2 you can save yourself typing by replacing the call to the SuperBasic function JOB_NAME$, at line 870, with the Toolkit function JOB$, which expects the same parameters but returns nothing for SuperBasic and nameless tasks.

MAP_BAS assumes that system variables are at address 163840, and may need tweaks to run on obscure systems such as the Thor 20. Dual-screen Minerva is supported, as the program checks for VER$ "JSL1" and uses the extra Minerva parameter VER$(-2) to find the system variables. For maximum reliability, use the DIY Toolkit SYSBASE function rather than the SuperBasic version.

## PEEK$

The ubiquitous PEEK$ function is written out here in slow SuperBasic. Equivalent but much faster code is implemented in Turbo Toolkit and other extensions, such as the CGH and Quanta library toolkits. Turbo compiles PEEK$ directly into a call to its own run-time system, so the PEEK$ extension is not needed to run the compiled task. You should not use the SuperBasic function if a code version is available.

To save yourself yet more typing, and possible debugging, you can get the whole lot ready-compiled on disk. The SuperBasic, compiled task and resident procedure code for MAP appears in DIY Toolkit Volume H, along with heap extensions, device offsets, documentation and source for RESERVE, DISCARD and LINKUP, plus the original MEMORY_HEAP_BAS, DEVLISTS_BAS, and a set of stopwatch timing commands and functions introduced in QL World, May 1989.

DIY Toolkit volumes cost three pounds each, on our 3.5 or 5.25 inch disks or your cartridges (one per volume), plus four pounds per order for processing, so it pays to order several volumes at once. Write to DIY Toolkit, Cwm Gwen Hall, Pencader, Dyfed Cymru, SA39 9HA, or call Richard Alexander on 0559 384574 to place an order or obtain more details.

As ever I am grateful for ideas from readers, and welcome your suggestions and comments on past and future DIY Toolkit projects. Please write to me, care of Sinclair QL World, if there is anything new and useful that you would like to see implemented as a QL extension.



```
1220        REMark Is that an Ident Word (19195 = $4AFB) ?
1230        IF PEEK_W(jb_start+110) = 19195
1240          RETurn PEEK$(jb_start+114,PEEK_W(jb_start+112))
1250        ELSE
1260          RETurn "No Name"
1270        END IF
1280      END IF
1290    END IF
1300    RETurn "** Not Valid"
1310  END IF
1320 END DEFine JOB_NAME$
1330 :
1340 DEFine FuNction SYSBASE
1350 REMark Redundant if you have the DIY Toolkit equivalent
1360  IF VER$="JSL1"
1370    RETurn VER$(-2):REMark Locate Minerva system variables
1380  ELSE
1390    RETurn 163840
1400  END IF
1410 END DEFine SYSBASE
1420 :
1430 DEFine FuNction PEEK$(addr,len%)
1440 REMark Redundant if you have the Turbo Toolkit equivalent
1450 LOCal p,t$
1460  t$=''
1470  FOR p=addr TO addr+len%-1
1480    t$=t$ & CHR$(PEEK(p))
1490  END FOR p
1500  RETurn t$
1510 END DEFine PEEK$
```

# QMON 2

**INFORMATION**
**Program: Qmon II (2.05)**
**Publisher: Digital Precision Ltd., 222 The Avenue, Chingford, London E4 9SE. Tel: 081 527 5493.**
**Price: £39.95 (disk or mdv) plus 5% Europe, 10% to Rest of World.**

QMON was originally sold by Qjump, the firm run by one of the QL's main designers, Tony Tebby. Having moved to France, Tony has now arranged for this program to be re-released by Digital Precision in this new version (unfortunately there is currently no discount for users of the old version). Updates have mainly taken the form of ensuring that Qmon can be used in Minerva's two-screen mode, and also improvements for use in the Pointer Environment. There is however some suggestion that old versions of Qmon had trouble working on the Thor XVI, and it is not known whether this new version will now work on this.

*Qmon II* is a program for monitoring and debugging machine code programs and machine code SuperBasic extensions. It allows you to see machine code programs in action, examine their registers and code, and even alter parts of a program as it runs.

## Manual help

The program is supplied with a comprehensive A4 manual (which users new to machine code may find hard to understand, although the package is not really aimed at inexperienced machine code users) and a disk or microdrive containing the program and various demonstration files. The manual itself is not very different from the one originally supplied by Qjump. As with most of DP's manuals, it is more than adequate. It takes you through the demonstration files step-by-step, giving you an idea what Qmon is capable of, both in Mode4 and Mode8.

To load the program, you can either use *Toolkit II's* LRESPR, or the supplied boot program. Once loaded, there are two main ways in which you can invoke Qmon. If you wish to monitor a task, it is best to use the form 'QMON #3,flp2_named_task' (oddly enough for one of Tony's programs, this form does not currently support default directories, and so the full file name must be supplied). This proceeds to set up the given task file and then immediately returns control to Qmon so that you can monitor the task's progress.

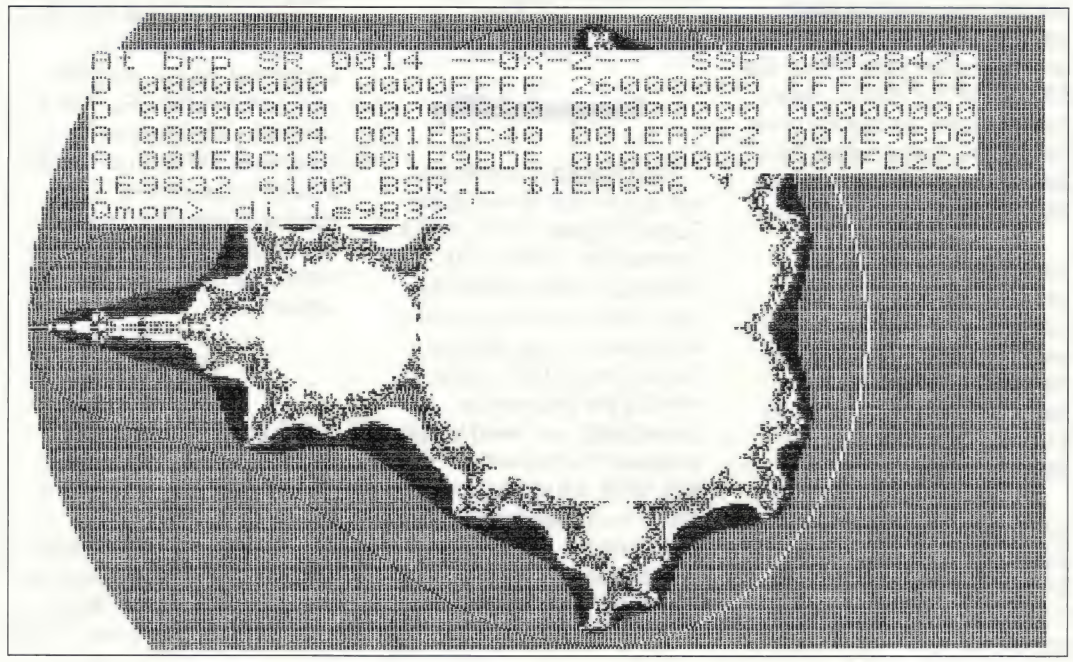
In this form, you can still use SuperBasic by pressing Ctrl-Space and then Ctrl-C. However, you may find that unless you have the pointer environment present, if you remove the task from memory other than from within the task itself (eg using RJOB), Qmon leaves a little cursor present on screen, which although it will not allow anything to be typed, slows down task switching as you will need to Ctrl-C through this as well. The pointer environment does however also create other problems as you may find that you cannot access SuperBasic while Qmon's cursor is active. When the program which is being monitored is removed, Qmon has no further effect until it is invoked again with the command QMON.

## Short commands

The other main way of invoking Qmon is to use the command QMON #ch, which attaches Qmon to the QL operating system. This means that Qmon will trap many different types of processor exceptions (such as Address Errors) and allow the user to regain control of the system (and possibly alter the offending machine code instruction) rather than resulting in a system crash.

Once invoked, Qmon prints the prompt 'Qmon >' to the

**Tony Tebby's QMon has been updated, and republished by Digital Precision. Rich Mellor assess the new version.**



given channel. You will need to type in short commands (normally one or two letters followed by various numerical parameters) to get Qmon to do anything. Qmon provides an extremely wide range of facilities for you to monitor machine code programs. Break points may be set either in rom or ram by using the command B address. When a break point is encountered while any program is running, the current program will be stopped and control returned to Qmon to allow you to inspect the machine code registers and much more. You can also disassemble parts of memory, and there is even a small line assembler included which allows you to alter given machine code instructions. If you are using a part of memory which has been set aside to be used as data, this can be displayed in both hexadecimal and Ascii, and you can then alter its value to test certain situations.

## Running trace

Other commands allow you to trace a program as it runs, either stopping every time that an instruction is executed, or only every so many lines. These trace instructions can even be used to allow a program to run as normal until a certain condition is met (or until a sub-routine returns). By using the LS command, you can even trace a program running in supervisor mode, although I have encountered some difficulties in using this, as the program tends to 'lock up' when the interrupts are disabled, or from within an SD.EXTOP machine code routine on my Minerva (1.91) rom. This may however be due to the fact that Minerva does not smash the trace flag on a TRAP #0 instruction.

```
1119/1440 sectors   Device is Read/Write  1992 Jul 18 17:07:46

Us      Userno.
Attr    RW Read/Write
        RO ReadOnly
        V  Visible
        H  Hidden
Vers    Version

Name                          Length Attr Us Typ  Update date          Backup date          Vers  Space

boot                             835 RW V  0 Dat  1992 Jul 18 13:30:00 1961 Jan 01 00:00:00    1      0
Ptr_Gen                        14638 RW V  0 Dat  1992 Jun 27 14:08:00 1961 Jan 01 00:00:00    0      0
WMan                           10126 RW V  0 Dat  1992 Jun 27 14:08:01 1961 Jan 01 00:00:00    0      0
Hot_Rext                       11772 RW V  0 Dat  1992 Jun 27 14:08:03 1961 Jan 01 00:00:00    0      0
Config                          7616 RW V  0 Exe  1992 Jun 27 14:08:04 1961 Jan 01 00:00:00    0   4000
Config_English                  6078 RW V  0 Exe  1992 Jun 27 14:08:05 1961 Jan 01 00:00:00    0   1792
Hot_Rext_English               11684 RW V  0 Dat  1992 Jun 27 14:08:07 1961 Jan 01 00:00:00    0      0
Ptr_Gen_Updates                  912 RW V  0 Dat  1992 Jun 27 14:08:07 1961 Jan 01 00:00:00    0      0
system_rext                    16508 RW V  0 Dat  1992 Jun 27 14:09:51 1961 Jan 01 00:00:00    1      0
SysRun_Rext                     4230 RW V  0 Dat  1992 Jun 27 14:08:11 1961 Jan 01 00:00:00    0      0
SysAttr_Rext                    5852 RW V  0 Dat  1992 Jun 27 14:08:12 1961 Jan 01 00:00:00    0      0
IODev_Rext                      1558 RW V  0 Dat  1992 Jun 27 14:08:12 1961 Jan 01 00:00:00    0      0
QIMIDev_Rext                     468 RW V  0 Dat  1992 Jun 27 14:08:13 1961 Jan 01 00:00:00    0      0
KeyReset_Rext                    430 RW V  0 Dat  1992 Jun 27 14:08:13 1961 Jan 01 00:00:00    0      0
KeyRecall_Rext                   364 RW V  0 Dat  1992 Jun 27 14:08:13 1961 Jan 01 00:00:00    0      0
German_Tra                       262 RW V  0 Dat  1992 Jun 27 14:08:13 1961 Jan 01 00:00:00    0      0
English_Tra                      262 RW V  0 Dat  1992 Jun 27 14:08:14 1961 Jan 01 00:00:00    0      0
MakeROM_Bas                      641 RW V  0 Dat  1992 Jun 27 14:08:14 1961 Jan 01 00:00:00    0      0
MakeROM_Txt                      807 RW V  0 Dat  1992 Jun 27 14:08:14 1961 Jan 01 00:00:00    0      0
System_Dir_Bas                  2169 RW V  0 Dat  1992 Jun 27 14:08:15 1961 Jan 01 00:00:00    0      0
System_Chan_Bas                 1290 RW V  0 Dat  1992 Jun 27 14:08:15 1961 Jan 01 00:00:00    0      0
System_Upper_Bas                 388 RW V  0 Dat  1992 Jun 27 14:08:15 1961 Jan 01 00:00:00    0      0
System_Mem_Info_Bas              703 RW V  0 Dat  1992 Jun 27 14:08:16 1961 Jan 01 00:00:00    0      0
System_FLP_Info_Bas              744 RW V  0 Dat  1992 Jun 27 14:08:16 1961 Jan 01 00:00:00    0      0
System_TRA_SET_Bas               490 RW V  0 Dat  1992 Jun 27 14:08:16 1961 Jan 01 00:00:00    0      0
System_UserDir_Bas              1307 RW V  0 Dat  1992 Jun 27 14:08:17 1961 Jan 01 00:00:00    0      0
For_C_IOFattr_Asm                743 RW V  0 Dat  1992 Jun 27 14:08:17 1961 Jan 01 00:00:00    0      0
For_C_IOFattr_Rel                132 RW V  0 Dat  1992 Jun 27 14:08:18 1961 Jan 01 00:00:00    0      0
HardTools_Rxt                    340 RW V  0 Dat  1992 Jun 27 14:08:18 1961 Jan 01 00:00:00    0      0
HardTools_Txt                    282 RW V  0 Dat  1992 Jun 27 14:08:18 1961 Jan 01 00:00:00    0      0
System_Updates_Txt              7757 RW V  0 Dat  1992 Jun 27 14:08:19 1961 Jan 01 00:00:00    0      0
Ing_TEXT_ext                   22360 RW V  0 Exe  1992 Jul 18 13:27:05 1961 Jan 01 00:00:00    0   4096
```

```
Example 1 - (see printout)
--------------------------

Example 2 - example of disassembly output of Qmon II
----------------------------------------------
1E98C4 47FA LEA $1ED0CC(PC),A3
1E98C8 4281 CLR.L D1
1E98CA 3213 MOVE.W (A3),D1
1E98CC D7C1 ADDA.L D1,A3
1E98CE 247A MOVE.L $1E9744(PC),A2
1E98D2 223C MOVE.L #$1FFF,D1
1E98D8 24DB MOVE.L  (A3)+,(A2)+
1E98DA 51C9 DBF D1,$1E98D8
1E98DE 7000 MOVEQ #$0,D0
1E98E0 4E75 RTS
1E98E2 45FA LEA $1E98EC(PC),A2
1E98E6 6100 BSR.L $1EA2CA
1E98EA 4E75 RTS
```

Addresses can be entered into Qmon in various ways, including by reference to the values of address registers used by the program, for example the command DI3 10(A1) 100 would disassemble one hundred instructions from the address contained at A1+10 onwards to Qmon's channel 3 (this enables Qmon's output to be sent to a printer or a disk). If you are not certain of the address required, Qmon also includes facilities to search in memory for either a set of bytes, a string of characters or even a specific machine code instruction.

## Complex tool

Qmon is an extremely complex piece of software and in such a short review as this, I cannot hope to comment on all of the facilities provided by Qmon, which include exception processing, transient windows in which Qmon can appear without destroying the screen underneath Qmon's window; the facility to recall a set number of machine code instructions which have just taken place; and even a macro command allowing you to tell Qmon to carry out a set sequence of commands at certain points in a program.

As with any programming tool, there are always various features which programmers would like to see (for example, a means of storing addresses), but it can be guaranteed that Qmon will contain most (if not all) of the monitoring and debugging facilities that even the best machine code programmer could desire.

# QL-System

**QL-System is designed as a software add-on to Toolkit II. Rich Mellor sizes it up.**

**INFORMATION**
**Program: QL-System**
**Publisher: Jurgen Falkenberg, Thanweg 36, D-7539 Ersingen, Germany.**
**Price: 78 DM + 20 DM (p&p) (approx £28). Add 15DM if not by Eurocheque.**
**Disk only.**

This toolkit is designed to act alongside Qjump's Toolkit II, and adds various additional facilities for disk and display handling to the QL.

The toolkit is supplied with various other miscellaneous programs to enhance the QL's facilities, together with an A5 manual in a plastic binder. The manual is quite well written, although it does tend to show its translation from the German in places and also fails to cover the additional programs supplied on the disk in any great detail (if at all). Although an index to the various commands is provided, in true *Toolkit II* style, this is by reference to the numbers of the sections of the manual. I have always found that page references are much easier to follow, but then I daresay there are pros and cons to both methods.

## Two groups

The commands can be split into two groups: those dealing with disks and files; and those dealing with the Qdos system.

The toolkit is initiated by the command

```
a=RESPR(17408):LBYTES
flp1_system_rext,a:CALL a.
```

The command SYS_EXT then links in the new com mands and activates Toolkit II (if available). Although Toolkit II is not neces sary to use the toolkit, it is advisable, as most of the commands require this to be present in order to work. Level-2 device drivers are also required by some of the commands - the index lists which extensions are required for each command to work.

The set of file commands allow you to add various attributes to each file. The most unusual of these is the command SET_FATTR, which allows you to specify whether a file is to be read only, hidden altogether or only available to certain users. For example, SET_FATTR \flp2_text,ru3 will make the file 'flp2_text' invisible to anyone other than user number 3, and in any case the file will be made read-only. The command USER allows you to alter the user number, which allows you to set up user areas, as only those files with either the same user number or a user number of zero will appear in directory listings or even be accessible (files are saved with the current user number).

## Password

The toolkit also allows you to set up Supervisor files, by giving them the user number 63. Such files are only accessible if you enter the correct password (this can be configured), however, in version 1.16 this password does not work properly unless you use the default password (this should however be quite easy to fix).

Although this is a good idea, the password is not integrated into the file, which means that anyone with their own copy of QL-System can access that file by using their own password (machine code programmers may also find it quite easy to avoid entering the password). Also, these file attributes will only have any effect on a QL on which the QL-System is present, or at least the driver. Although a further two cut-down versions of the toolkit are included in the package, the manual does not say whether these may be freely distributed with commercial software which wants to make use of these user areas.

New Basic commands also allow you to read the current user number and file attributes, and there is also a wild card command to allow you to set attributes on several files at the same time (making it easier to set up user areas). A new machine code TRAP #3 command is also included to enable these attributes to be set/read from machine code.

## Device commands

Commands are also provided to allow you to check whether a given medium is write-protected (although this does not work with microdrives); toge ther with commands to return the cluster size of a disk, the number of available and free clusters, the length of a file header, the drive number and original medium name (useful after WIN_USE commands), and the name given following a FORMAT command.

Other file commands require the presence of Level-2 device drivers (such as on the Gold Card), and generally echo commands provided on those device drivers, namely functions to set and read file backup-dates, version numbers and file updates. Gold Card users will have come across these already, although oddly, there are also wild card variants of these commands in this toolkit. Again machine code TRAP calls are also added to complement these commands.

Some users may also have noticed that although Toolkit II provides functions to read the file type, dataspace and extra information stored in a file's header, there is no way of setting this information. QL-System now in cludes the commands SET_FTYPE, SET_FDATA and SET_FXTRA to fill this gap (together with the associated wild card versions of these commands). Other commands allow you to produce alphabetical directories (ADIR and ASTAT) and also to set both the data default and program default at the same time (DIR_USE).

## Qdos set

The Qdos system set of utilities include Job and Channel control commands; commands for using the pointer interface and the hotkey system; and general utilites.

The Job control commands

allows the same parameters as Toolkit II users will already have come across. QL-System however, provides commands which are sorely lacking from Toolkit II, namely commands to suspend Jobs and release them; and functions to return the identification number and start ad dress of the given Job. Channel control commands will print out a full list of all currently open channels, together with the ID of the task which owns them; as well as allowing you to close any given channel from any task. Again, func tions return the address of the channel definition block; Qdos channel identifi cation number; number of bytes per screen line (useful for checking if a window has been opened in extended mode 4 on the ST-QL emulator); and much more.

If you intend writing a program which may be used under the pointer interface, a function is even provided to check whether or not output on that channel is possible (ie whether it is at the top of a pile).

## Pointer commands

Many of the pointer and hotkey interface commands make use of the various additional TRAPs provided by these systems, and include the ability to store windows, set the stuffer buffer and mouse speed, find Thing version numbers and find the outline limits of a given channel. Other facilities allow you to list all currently available devices, return the current screen mode and address of the system variables and even the current baud rate. Some of these commands are based upon similar commands which have appeared in *DIY Toolkit*, such as commands to SET and CHANGE absolute variables so that they can be used much as the PI function built into the QL. Many of the other commands can be found in various public domain libraries, although it is always handy to have them all in one single toolkit.

Some undocumented Basic programs on the disk will provide full de tails about the contents of a disk, details of a given channel, details of memory usage, and link in a German translation table.

An updates file (sadly still in German) goes on to explain that the public domain file IODev_rext is also supplied on the disk, which adds named pipes, a nul device, and a MEM device to the standard QL setup. The file KeyReset_rext installs a small machine code task that resets the QL whenever the keys Ctrl-Alt-7 are pressed; and a couple of other files which I am not certain what they do... There is even a program included to make the toolkit rommable if you have access to an eprom programmer.

## Configured

Various parts of the toolkit can be configured using the standard CONFIG program supplied, enabling you to alter the size of the windows set when the WSET command is issued (similar to WMON/WTV). For those of you who do not have the pointer interface, the latest version is also included, although there is no documentation on this - for this you will need to purchase Qpac II or the Qptr manual (although I would recommend the former as an introduction to the package).

Overall, the toolkit contains lots of useful little goodies, but is perhaps slightly overpriced, as many similar commands can also be obtained from the public domain libraries. I would however have liked to compare the package with Toolkit III produced by Digital Precision, as I believe that this may also include similar utilties to the file attributes.

# PRINTER REPORT

**Alex Munden set up shop with the Panasonic KX-P1080 many years ago, and still finds it suits his business.**

The Panasonic KX-P1080 is not a new machine, having been superseded in 1987 by the 1081, which has also (recently) been superseded by the 1170. These are narrow-carriage, 9-pin models priced at under £200. The KX-P1080 still turns up on the second-hand market, and refurbished examples are occasionally available from well-established suppliers.

Eventually, I made the rash decision to start my own business, and it became apparent that I would need a 'professional' machine. Shortly afterwards, Sinclair reduced the price of the QL, and I decided to tkae the plunge. Two replacements later, I had a working system, with a black and white TV monitor, and a Brother M1009 for hardcopy.

## Better output

As the business developed,I needed to improve the look of my output to Near Letter Quality. I found the Panasonic KX-1080 on prominent display in a shop window. Having solicited a demo of the printout quality, and assured myself that ribbons and things were available, I waved a card and walked out with it.

The KX-P1080 is an Epson-compatible, 9-pin, 80-column dot matrix printer with a parallel interface as standard, although an optional RS232 serial board can be fitted. As I already had a Miracle Systems Centronics adaptor, the standard interface was sufficient, and it ran as soon as I plugged it in.

Draft output of 100 characters per second, and a somewhat slower 20 cps in NLQ mode, is adequate for my needs, and I am partiularly impressed with the NLQ output. (So am I - Ed.) There are true descenders (the bits below the line in lowercase letters g,j,p,q and y) in all print modes, and printing is bi-directional - the print head works in both directions. It is also logic-seeking, skipping blank spaces at the beginning and ends of lines and between paragraphs. Print operation is not excessively noisy, although I would not volunteer a serious telephone conversation of any length with the printer running in the same room. It can be distracting.

## Form widths

Friction and tractor feed come as standard, and are selected by a lever to the left of the casing. The friction feed sprockets are fully adjustable for different form widths, and also serve as a paper guide for cut sheets.

Physically it has a small(ish) footprint (403 by 286 by 115mm), although it requires more space if continuous forms are used, and it is solidly built at 6 kg. A wire frame paper guide is supplied for continuous stationery but there is no provision for a cut-sheet feeder.

Both the print mode selector switch and the paper feed switches are on top of the casing, and the dip switches for setting the machine's defaults, such as page length and character set, are reasonably accessible under a clear cover below the print head guide bars. To the left of the platen the print head pressure lever to handle up to three sheets of paper.

The only available typeface is Courier, with type pitch sizes restricted to pica (10 cpi), elite (12 cpi), semi-compressed (15 cpi) and compressed (17 cpi). Elongated characters are also available in all pitches. In NLQ I have seen worse results from more recent and expensive PC printers. Proportional spacing is supported in pica pitch but not in elite. The type styles include superscript, subscript and italic, although neither superscript nor subscript is available in NLQ. With the international and italic international character sets, there are nominally twelve print modes .

```
Panasonic KX-P1080 print demo

Draft Pica pitch - 10cpi
Draft Elite pitch - 12cpi
Compressed mode - 15cpi
Pica double width
Elite double width
Superscript characters

Subscript characters
Draft italic script
NLQ italic script
NLQ Pica pitch - normal spacing
NLQ Pica pitch - proportional spacing
NLQ Elite pitch
NLQ compressed
```

Of the ten international sets on the UK model, seven are switch selectable and three by software. (These figures will be different for more recent models, of course.) Apart from USA and UK, there are alternative dedicated characters for French, Danish (two sets), Swedish, Italian, Spanish, Japanese and Norweigan. Most of the international character sets are not very useful in printing in English, but do provide access to the pound, dollar and hash symbols, as well as some lesser-used ones. In all, there are 256 printable Ascii characters, from 0 to 255.

## Still Quill

Despite recent updates in wordprocessing, I still use my Quill, enhanced by Athene Consultants' TurboQuill+, for nearly all my work.

Graphics output is good, with minimal 'streaking', even from the standard Easel GPRINT printer driver. I have also had impressive results from Page Designer 2 and Omindump, which I use to demonstrate layouts when designing forms and publicity material for clients.User definable graphics are also supported, and there is a downloadable graphics mode. 8-pin and 9-pin bit-image graphics are available in standard, double and quadruple density.

The ribbon cartridge from the makersr has a nominal three-million character life, although this is reduced by heavy graphics use. There is also a supplementary ink reservoir, which can be activated by poking a pencil into a hole when the quality starts to fade. The ribbon is not re-inkable, and I would not recommend any of the re-inkable substitutes.The print head does a good job of shredding ribbon seams with heavy use. The manufacturer's own ribbons are good, consistent, and readily available from Microsol Ltd.

On the down side, the KX-1080 does not support paper parking or scroll back, and will only accept 11 in or 12 in lengths. This can be irritating when using continuous stationery, not being A4. Also, when using continuous paper, the Centronics port is just below the paper path. The Miracle adaptor adds nearly three in to the back of the casing, and can cause the paper to drag. The tractor feed is not really suitable for NLQ output. However, the tractor sprockets can still be used in friction mode, and serve to keep the paper aligned on long print runs.

Only 1 KB of buffer memory is supplied, and this is not expandable, which means that a print spooler is useful to prevent Quill or Archive from being tied up during a long print session.

To sum up, the printer has been a good, reliable workhorse, requiring very little maintenance, and it is still well supported. There are more up to date and, by now, better-specified printers on the market, but for someone with a limited budget, a KX-P1080 is well worth considering.

# Perfect Paragraphs & Shining Characters Part 2

**Martin Harris continues with his deportment course for properly proportioned printers.**

In last month's instalment of this article, I ended by stating that the next task is to use the programming structure we had established, to keep track of where our document is as regards printer modes, and also to perform some simple automatic error corrections.

## Modes and errors

For this section we must first have established the control codes for the various print modes. These are to a large extent random. (See the final section of the printer driver in figure one.)

We shall start our system by tracking the status of italics, bold, underline, proportional and superscript. It is very simple to switch the toggles on and off (lines 3100 to 3140). Once we have kept a track of which printer modes are in operation in this simple way, we have most of the background information necessary for detecting and correcting our various errors. We don't have to do any more than this for the time being, because the main error that can happen is that an end-toggle has been omitted and, since we cannot test for something non-existent, we must come back to this point only at the place where the close toggle should have been, ie at the end of a paragraph or line. Our trick will then be to check at those points if the relevant toggle status is on and, if so, either to have the program insert the close toggle at the end of the paragraph if we are sure that in every such case it is an error, or to generate a report so that we can decide ourselves whether it is an error or not. For example, we may want bold to span several paragraphs in certain situations, but usually we would have closed the bold toggle before a paragraph end. We shall come back to this operation in more detail.

## Tackling toggles

However, with certain toggles, an error may have occurred simply by switching on or off, so it is here, when we first encounter the toggles, that we must deal with any likely problems.

Our first probable error is that a change of density toggle occurs at the start of a paragraph. The sequence:

$\overline{Q}$The start of a paragraph<br>followed by the next line$\overline{Q}$

is wrong because the spaces at the start of the line before the Ctrl-Q are still in the old density, whereas the spaces at the start of the second line are in the new density. The result is printed like this:

The start of a paragraph
followed by the next line

The programming solution (line 3170) is based on keeping track of the column number on the line (the variable "tab_posn", which is reset to zero each time an end of line is reached) and produce an error if the change of density occurs after the start of the line. But in that case, it will always be an error wherever the code appears, therefore we must allow for a change of density mid-way through a line without that being an error. Hence the extra condition that the error must occur before the first character on the line (kept by the array "info_array%"). You may, however, want the strict condition that an error is reported unless this type of printer code occurs at the very end of a line.

## Pica toggle?

The same error is generated (lines 2920/2930) for the compressed toggle but not for the pica toggle because there is no such thing! Both the printer and this program default to pica. If, however, you always use elite (and start your document with the elite toggle) you will no doubt want to change this so that the same error is generated when the elite is switched off, rather than on (it will effectively become an on-pica toggle). The physical resolution of this type of error is to move the toggle to the blank line above or to the end of the previous text line.

Since we are devising a program structure that moves largely sequentially through the text, it is somewhat inappropriate to have the program correct this error, as it would involve going backwards through the file, inserting a character, then returning to the main file pointer position, an overly complex operation. Therefore we should simply generate an error and correct this manually afterwards. We shall deal with the error reporting separately. In any case, we should also manually check that the left margin is correct when changing density. If you want the left margin of elite text to line up exactly with a prior paragraph of pica, only the following settings will work:

Pica margin: 5 10 15 20
Elite margin: 6 12 18 24

and so on. Another error likely to occur is with enlarged density. Despite our perfect knowledge of our word processor and printer driver (!) we may forget that we will only be allowed half the number of enlarged characters on a line. We will need to check when we come to an 'off' toggle how long the line was; which means that, when we encounter an 'on' toggle we must also first have made a note of its position on the line. Let us call such a marker "enlstart"; when we reach a condition to set the toggle to on, we simply assign the value "tab_posn" (see above) to this variable (line 3230); upon arrival at the off toggle, we must decide how an error condition is to be generated. The characters up to the on toggle are not doubled, so the correct formula is of course:

characters up to on toggle
plus twice length of enlarged text

A little algebra converts this to

2(tab_posn) - enlstart

(tab_posn being the position of the off toggle). We do not have to be precise about the length of line that this amount is not allowed to exceed, since we shall merely report an error

and review manually afterwards, so let us take 80 as the limit. Thus we have the error condition:

```
IF (2(tab_posn) - enlstart) > 80 THEN report_error
```

Once the programming structure has been established, many other such situations can be relatively easily picked up and dealt with even for much more advanced printers than mine.

## Hierarchy

We must still deal with the printer hierarchy, which promises to be slightly more complex.

The main problem that I encounter is that elite and proportional type are incompatible with compressed. I also like compressed to be double strike, so that it is not dotty, and this may create other incompatibilities.

We may deal with this on the basis that compressed is always compatible with pica and non-proportional, so if we ensure that whenever compressed is used, the printer status reverts to these, we shall overcome our problem. We shall call the elite status before the compressed toggle was encountered "precompel". Similarly for proportional, "precompppnl".

At the start of compressed type, the following operations are needed:

```
IF elite is on:
set precompel to 1,
otherwise 0
switch to off by inserting
Ctrl-Q into the text

IF ppnl is on:
set precompppnl to 1,
otherwise 0
switch to off by inserting
Ctrl-P into the text
```

At the end of compressed text we must:

```
IF precompel is 1:
insert ctrl Q to get back
elite
IF precompppnl is 1:
insert ctrl P to get back
ppnl.
```

We must also take care over the order in which we make any insertions, to avoid the original problem recurring. (For the above operations, see lines 2920 to 3050.)

This does not overcome every hierarchy problem (for example, bold and NLQ are sometimes incompatible with other modes). Some such may occur very rarely, depending on your own preferences and needs. If in doubt over the printer manual's interpretation, experiment with some random text and various combinations of control codes, then add your own program lines accordingly.

Note that, depending on your preferences, some of these problems may be more simply countered by altering DIP switch settings on the printer itself. If you do this, remember that the converse problems will occur unless you absolutely never need these other combinations of print modes.

**The last of the "simple" areas is that of our backspace-overprinted characters, which must not be used with ppnl (lines 2900/2910).**

You have probably got the idea loud and clear by now: if ppnl is on and you need a backspace-overprinted character: insert an off-ppnl control code, print the character (to which the printer driver will add the backspace and overprinted accent), then insert an on-ppnl control code afterwards. A slight snag arises here in connection with other work being done by the program to do with proportional insertions.

For instance, consider this example of text:

```
P(i)lê tre un aliné a:

 PThe cat sat on the mat
All that glitters is either gold
or not.
```

Which will print like this:

```
(i) être un aliné a:

The cat sat on the mat
All that glitters is either gold
or not.
```

*[Note: in Martin's original example, the 'cat' rhyme is on the same line as the accented foreign words. The point is that the embedded codes offset the rhyme slightly to the right but, when they print, the 'cat' and 'gold' lines line up exactly on the left.]*

We will get the computer to insert the Ctrl-Ps for us later in the article. Notice, however, that between two Ctrl-Ps, it would be wrong to insert further Ctrl-Ps. I monitor this situation as "status6". Similarly, compressed must be off before attempting this operation.

## The main loop

Having impatiently gone ahead to look at some simple status keeping and error corrections, we should now return to the main structure, in which, as we have already mentioned, we must keep tabs on the line number (for error reporting) and the position of the character on the line, (for various purposes). We must also trap the end of line and end of paragraph points since it is here that we have decided that most of our end toggle errors will occur.

Do not worry about the line_analysis loop (lines 1750 to 2400) for the time being, as it is part of the proportional insert system which we shall discuss later. Our main loop therefore starts at line 2450.

The first thing is to set all our variables to zero before entering the loop (lines 2410 to 2440). The line number itself of course starts at 1, not 0. "number" and "count" (line 1720) are not set to zero but to the start of memory where they are or will be stored. These will be differently expressed if you are not using memory but disk, as explained earlier.

By the way, don't even think about using microdrives for these operations! They just won't stand the pace.

Some of the variables are for proportional inserts so forget them for now - all will be revealed in due course.

## Exotic names!

Without further ado, we start our loop. I chose the name "loop" because it sounded so exotic. Actually, you can assign any name to loops, precedures, variables, and so on, of whatever length without slowing your program down since, when it is compiled, these will all disappear and become mere numbers.

Inside the loop it is essential to set the conditions for exiting from the loop in the very first instance, and not at the end, (line 2470) otherwise the loop will cycle one time more than you want, which may have dangerous effects. The point of poking 10 (end of line) at the end is to overcome a bug that developed, whereby the last line of a document was never printed. I still don't know the cause but this is my solution. If anyone can shed light on the cause, please write to QL World. Any ideas?

Let us remind ourselves of our simple structural rules before continuing: each time round the loop we must increment the variables "number" and "count", the source and destination files respectively, and whenever an additional character is inserted into the document, "count" must be further incremented so that, when the process is finished, we know what section of memory to save to disk.

## Buffer space

In this respect, note that the buffer for this purpose (line 1700) was set to 1.05 times the size of the source file, precisely to allow for further characters to be inserted. The program would still work without this, but it would be very risky, since it may be poking into areas of memory already used for something else. When checking the medium, this length was also assumed in order to check in advance whether there was enough space (line 1650).

We should also increment the position of the character on the line (lines 2570/2580)

but not for control characters since, when printed, they will take up no space, and additionally ensure that when an end-of-line is reached, the line position is reset to zero and the line number is incremented (lines 2670/2680).

To correct underline errors will be the next task. If an underline code occurs without being switched off before the end of a line, it may be an error or it may be that the final off switch is later in the same paragraph. In either case, a fresh underline must be inserted at the end of the line. We shall consider that if a manually inserted underline code is found before the end of the paragraph, no error is generated. In this case, we must get the computer to insert the underline characters at the end(s) of the line(s) and the beginning of the following line(s), until it reaches the manually inserted one. If however, it does not reach such a code by the end of the paragraph, it must both insert the extra code at the end, to avoid drastic misprints, and generate an error condition.

The program must therefore be in two parts: firstly, it must detect an end of line with underline status on and add the extra code. Secondly, it must insert a second code on the next line at the start of printed text (and not before).

Since we certainly do not need an underline code at the start of every line, we need some way of remembering whether the computer was in the middle of correcting an underline. We of course do this at the end of the line when we add the extra code. To summarise our logic:

**At the end of a line: if underline status is on, add an off toggle and remember that we had done this (lines 2720/2730).**

At the start of a line: at the first printable character, check our underline correction "remember" status (line 2590) and if it is on, add an extra underline toggle.

There are lots of things that our program has to "remember" in this way, and I have simply called them "statusn". In the case we are dealing with now, I called it "status7". I could have called it "in_the_middle_of_an_underline_correction" but opted for something a little shorter.

## Reset to zero

When such a condition is met at the start of a line, the variable "status7" must be reset to zero before the next character on the line is examined, and the underline status itself must be changed in case the program loop discovers on its travels a manually inserted off-toggle. Finally, end of paragraph matters can be detected and dealt with simply by looking for an end of line (SELect on n1 = 10) followed immediately by another end of line (IF PEEK(number+1) = 10). If this situation is encountered while status7 = 1, then an error must be generated (line 2740). Also, status7 is reset to 0 because the computer has now physically dealt with the error so we are back to a normal situation.

The remaining end-of-paragraph type code errors are all dealt with in a similar way, except that nothing is actually corrected; only error reporting is performed. This could be a double-edged sword: if you switch the actual status off, without physically adding the end toggle, error reporting can continue correctly since, when the next paragraph to start with a toggle occurs, the program will think it is a start toggle and report an error if that paragraph does not contain a second toggle, and so on correctly, whereas had the toggle status been left as it was in reality, the program would think the next toggle was an end toggle such that a missing toggle later would not be reported and the main purpose of our program, to report errors, would have been defeated. (If that's perfectly clear.)

## Checks stymied

However, the rub is that if the status is altered from what it really is, no further hierarchy checks can be valid, which is especially upsetting if the actual end-toggle occurs deliberately later in the document.

**For this reason, error detection statuses must be separated from the actual statuses, if the program is not to physically correct the error.**

Hence distinct statuses (eg italerrstat) are used, both to to detect missing end toggles at end of line or paragraph (line 2770) and at the point where the real toggle status is updated (line 3100) and thus the real status and the theoretical error status are kept independant from each other.

In this way, when an end-toggle is detected as missing at the end of a line or paragraph, the next toggle to occur on its own in a paragraph will generate a second and corresponding error, and no action need be taken if that was what the user intended. If however the number of such errors is not an exact multiple of two, then a real error must have occurred. This situation can and should also be reported.

It is a matter of choice what to physically correct and what to merely report. I chose only to correct underline, in order to allow other print types run over several paragraphs. The computer will then report errors, but not correct them, leaving me to make the choice. As a matter of detail, you can see that I wanted superscript to generate an error if the off-toggle was not encountered by the end of the line rather than end of paragraph (line 2710). Furthermore, enlarged mode must be switched off at the end of a line as instructed by the printer manual (line 2700).

You now have examples of all the printer type errors I can think of, and a structure which renders further types of error-checking and physical correction particular to your printer easy to add. You might care to extend the hierarchy checks if you are in the habit of switching between densities, etc., often, perhaps to add more generalised line length error reports or even page length errors if you use different spacing and want to ensure that you have neither large blanks at the end of a page, nor accidentally run over.

## One can dream

I also thought, perhaps over-ambitiously, of a fully-fledged interactive micro-justification system that would work by counting the exact number of pixels on a line, working out optimum line width, and replacing space characters by graphics spaces of varying widths. Of course, the program would have to be fed information on all the densities available to each printer, whether italics, proportional, bold, elite, pica, condensed, NLQ, draft, etc. The printer would then work more slowly, but nowhere near as slow as it would in full graphics mode. So far, there has not been a printer driver which allows enough information-bytes per "translate": to send a quad density graphics character to the printer takes nearly 30 bytes, while *Editor's* printer driver only allows 16. *Perfection Special Edition* does seem to overcome this, so I might give it a whirl one day.

*In next month's instalment, I will move on to choosing the types of errors which are best reported by the program, and whether to correct by hand or via the program.*

## MEDIA MANAGER SPECIAL EDITION
## MEDIA MANAGER

Media Manager Special Edition (MMSE) is a program to be used both when things have gone wrong as well as when things are perfectly OK. It allows for automatic, semi-automatic and manual correction of a huge variety of disk and tape problems. It allows you to explore disks and tapes to your heart's content, producing all sorts of different diagnostic reports. MMSE is very simple to operate, being menu-driven and assuming no degree of computer knowledge whatsoever.

MMSE also allows you to tidy, catalogue, sort and order your disks and cartridges.

The standard Media Manager is both less powerful and less user-friendly, but manages to work on an unexpanded QL.

Both programs allow for data transfer between PC and QL. With MMSE, this transfer is at file and directory level, is bi-directional and is completely automatic.

## SPECIAL DESKTOP PUBLISHER
## DESKTOP PUBLISHER

These programs are quite primitive compared to Professional Publisher. However, if you have not experienced that program as yet, you will find both of these very competent. Both are capable of producing excellent results. The cheaper one has fewer features but is able to run on smaller systems.

## EDITOR SPECIAL EDITION
## THE EDITOR

With the sole exception of Perfection, this is the best word handling system on the QL. Editor's features include an unrivalled degree of programmability and the ability to cope with the entire 256 character ASCII set. The Special Edition has enhanced document-type facilities, including column blocks and on-screen page break displays. Neither program is suitable for computing novices. Until Perfection, Editor Special Edition would have been our 'Desert Island Program'.

Editor SE can do a few things that Perfection can't, so the ideal combination is to have both (they are compatible at file level and can multitask). If you order Editor SE at the same time as Perfection, you can have Editor SE at half price.

## PROFESSIONAL ASTROLOGER
## PROFESSIONAL ASTRONOMER

The Astrologer program teaches you Astrology from scratch and enables you to automatically produce text narrative on personality delineation, year-to-year and minute-to-minute life predictions, compatibility interpretations and so on. Whether or not you believe in astrology – indeed, especially if you do not – this program is one that you cannot afford to have. You can tailor the readouts (both in terms of quantity and what is said) to your own particular requirements. The amount of fun you can have with this program is endless. Do not blame us if you start believing in astrology, though!

Astronomer is an extremely fast and accurate solar system calculator, with planetarium views, planet faces, eclipses, cinerama display etc..

## TURBO BASIC COMPILER

Turbo is the finest BASIC compiler for the QL and arguably the finest BASIC compiler for any computer!

Turbo automatically converts working BASIC programs into optimised machine code, usually with no need for human intervention. The benefits of this conversion are vastly enhanced running speed (as well as much faster loading, encryption and automatic bug fixing for a variety of QL interpreter oddities). Typical speed-up is 40x – 100x.

Turbo is provided with a 200 command toolkit, adding many useful commands to BASIC. Most of these commands will be of immediate use to the programmer, whether he is a novice or an expert. There are commands to load strings and floats into RAM, and to extract them automatically; to search memory and to move its contents; to control jobs and change their priorities, manage pipes, allocate and deallocate memory, to control both rubber and virtual arrays, to present INPUT with an editable default, to have random access to files and much more.

## TOOLKIT III

Toolkit III starts where Toolkit II stopped, adding about 60 new commands and enhancing many existing dual functions. Toolkit III is available either on disk or on ROM, and works whether or not you have Toolkit II.

Toolkit III commands can, with only a couple of exceptions, be compiled using Turbo.

## QFLICK CARD INDEX

All QL owners have a copy of Archive, supplied free with the QL. While Archive is competent, it is very hard to get to grips with and is not particularly fast. QFlick presents a very convenient alternative – a snappy , simple-to-use, pointer-controlled card file database. You can move data between QFlick and Archive in either direction.

QFlick is not itself programmable but we document its data structure and give guidance on how to program it using Turbo.

## ARCHDEV + RTM
## DATABASE ANALYSER
## ARCHIVE TUTORIAL
## NAMES + ADDRESSES
## MAILMERGE
## DAT–APPOINT
## SEDIT
## SCREENPRINT
## RECOVER

This suite of utilities will greatly enhance your use of the Archive database system.

Archdev + RTM is a straight replacement for Archive: it gives enhanced speed, greater workspace and a much cleaner boot-up. All your existing applications will work.

Database Analyser provides very fast and comprehensive statistics about your Archive databases.

Archive Tutorial proceeds systematically through the whole philosophy and grammar of Archive, providing you with expert and patient guidance.

Names + addresses, Mailmerge and Dat-Appoint are ready-to-run, off-the-shelf Archive applications, providing an address database, mailmerging and appointment diary respectively. You now have no excuse not to use Archive.

SEdit allows you to create and edit screen format files in Archive. Screenprint allows you to print them out.

Recover allows you to get back lost Archive databases, created when you switched off the computer without properly exiting from Archive.

## XREF
## SUPERBASIC MONITOR
## BETTERBASIC EXPERT SYSTEM

XRef analyses the structure of a BASIC program, providing detailed reports on things like variable usage, what calls what, dynamic call hierarchy of procedures and functions, and so on.

SuperBasic monitor actually monitors and reports on the performance of BASIC programs as they run under the interpreter.

BetterBasic analyses and automatically corrects structural flaws in your programs and allows you to customise things like indentation, number of statements per line, filtering out of noise words, etc.

The three programs together provide a matchless diagnostic and auto-correcting facility for BASIC programs.

## TRANSFER UTILITY

This program copies files at high speed between devices, performing translates as it goes along. Ideal for all sorts of applications, including transfers from microdrive to disk.

## QMATHS SYSTEM

This is an incredible mathematical compendium for the QL. Pride of place goes to the symbolic problem solver: this can solve equations, simplify expressions, factorise, expand, etc, all symbolically. If you could sneak this one into a maths examination, you would have a formidable ally. QMaths knows about all the algebraic operators, powers, roots, brackets, trigonometry, matrices, determinants, vectors, factorials, permutations, combinations, binomials, exponentials, logarithms, hyperbolics, inverse functions, infinite series including Taylor & Maclaurin expansions, complex numbers, conversions, Fourier series, and lots of calculus: both differential and integral, including integration by parts and definite integrals. QMaths optionally displays its workings and comes with a superb interactive tutorial.

The package also contains an interpretive, fractal, image-generating language with loads of beautiful fractal programs supplied for you to use and edit – no programming skill is required.

There is also a multiple precision floating point maths package, giving calculations at precisions up to over 600 decimal digits of accuracy.

There is even more to this system, but we think we have told you enough.

## QMON MACHINE CODE MONITOR

The latest version of Tony Tebby's superb monitor: an absolute must for those who really want to know what is going on in the QL. No other machine code monitor even comes close.

Do not confuse this program with SuperBasic monitor, which monitors SuperBasic, not machine code.

## COMPARE

This program compares files – data or program – at colossal speed. Where a mismatch is detected, the relevant areas are highlighted and you can shuffle, displace and align very easily.

## CASH TRADER WITH ANALYSER
## PAYROLL

Cash trader with Analyser is an accounts system designed by businessmen and not by wretched accountants! Consequently, it has excellent reporting and management facilities, and is very flexible. It is aimed primarily at the layman, probably a sole trader running a small or medium sized business. All the features you would expect – including audit trail – are present.

Payroll is a reasonably flexible system designed to automate the payroll function in small businesses.

Both programs are configurable, with editable defaults letting you adapt the programs from year to year.

## HARDBACK WITH FINDER

This is the ultimate hard disk backup and management utility, with all the sophisticated features you could want. User dialogue is via overlapping pop-up windows – the whole program just feels right. It is possible to scan the disk at great speed, too.

## DISKTOOL WITH QUICKDISK

This permits you to add password protection to disks, to optionally increase disk storage capacity on DSDD drives by 36K and to increase speed of access by as much as 30%. All this is done while maintaining full compatibility. Automatic file management is also provided.

## DIGITAL C SPECIAL EDITION
## DIGITAL C

These are extremely fast and efficient C compilers, complying with and surpassing the Small C definition. The Special Edition goes much further, including support for structures, pointers, long pointers, >64K code size, direct access to QDOS traps, etc. The Special Edition C generates code that runs about twice as fast as the other.

## SPECIAL DEALS

5% off total if you buy 2 programs/upgrades;
10% off 3; 15% off 4; 20% off 5; 25% off 6+
Upgrades cost difference in price + £10
Non-UK Europe add 5%, rest of world 10%

For full terms and conditions, please refer to any of our QL World ads from Jan–Nov 1990, or write in including a SAE

## CPORT IMPROVED VERSION

A brand new CPORT system, enabling you to rapidly convert your SuperBASIC programs into C (ANSI or Lattice). The new (October 1992) version is now as close to being fully automatic as makes no difference – you must get it!

Owners of our earlier CPORT versions should return disk + SAE for a free upgrade.

## SUPERFORTH COMPILER WITH REVERSI

Forth is the most logical computer language. This compiler produces multitasking code. The manual teaches you Forth-83 from scratch.

## IDIS SPECIAL EDITION
## IDIS

These intelligent disassemblers make the otherwise terrifyingly complex task of understanding other people's machine code programs absurdly easy. The SE version, which has a higher hardware requirement, sorts out some routines, replaces addresses with names, untangles data from code and much more.

## QKICK FRONT END SYSTEM

This is a simple, easy-to-master, pull-down menu controlled multitasking front end. QKick runs in the background and can be called up at any time. It provides you with notepads, sophisticated file/sector/RAM handling, backing up facilities, a clock, diary, calculator, mini-database and so on.

## ADVENTURE CREATION TOOL SPECIAL EDITION

ACT is a must for every programmer. The name of the program is misleading, insofar as it has capabilities far beyond the 'mere' creation of adventures. ACT has utilities providing animated graphics, data compression, language design, parsing, maps, object-oriented control etc. If all you want to do is generate adventures, though, you do not need to be a programmer to use it. This is a purchase you will never regret.

## PEDIT

A fast, modern and capable printer driver for the programs bundled with the QL.

## MICROBRIDGE

Superb contract bridge bidder (ACOL etc) and player, using millions of random but reconstructable hands. Microbridge also includes a state of the art interactive bidding tutor and a clear instruction manual. There is nothing like this anywhere else!

## SUPER ASTROLOGER

A very cut-down version of Professional Astrologer – still great fun, though!

## SUCCESS CP/M EMULATOR

Allows your QL to run CP/M programs at great speed.

| | £ | |
|---|---|---|
| 3-D PRECISION CAD SYSTEM | 49.95 | d |
| ADVENTURE CREATION TOOL SPECIAL EDITION | 49.95 | e |
| ARCADIA GAME | 9.95 | f |
| ARCHIVE DEVELOPMENT SYS + RUN-TIME MODULE | 29.95 | a |
| ARCHIVE TUTORIAL | 19.95 | a |
| BETTER BASIC EXPERT SYSTEM | 24.95 | a |
| BLOCKLANDS GAME | 9.95 | f |
| CASH TRADER v3.3 + ANALYSER | 99.95 | c |
| COMPARE FILE UTILITY | 19.95 | a |
| COPY UTILITY FOR BACKING UP | 9.95 | a |
| CPORT BASIC TO C TRANSLATOR WITH C68K COMPILER | 99.95 | c |
| CPORT BASIC TO C TRANSLATOR | 89.95 | c |
| DATABASE ANALYSER FOR ARCHIVE | 19.95 | a |
| DAT-APPOINT APPOINTMENT DATABASE SYSTEM | 19.95 | d |
| DESKJET/LASERJET DRIVER FOR PRO PUBLISHER | 19.95 | c |
| DESKTOP PUBLISHER SPECIAL EDITION | 39.95 | c |
| DESKTOP PUBLISHER | 24.95 | d |
| DIGITAL C COMPILER | 29.95 | a |
| DIGITAL C SPECIAL EDITION | 49.95 | a |
| DISKTOOL WITH QUICKDISK | 19.95 | b |
| DROIDZONE GAME | 9.95 | f |
| DR-DOS v6.0 STATE OF THE ART PC OPERATING SYSTEM | 79.95 | e |
| EDITOR SPECIAL EDITION | 49.95 | d |
| EDITOR | 29.95 | a |
| EYE-Q GRAPHICS SYSTEM | 39.95 | a |
| FONT ENLARGER | 19.95 | e |
| GAMES COMPENDIUM (ALL FIVE GAMES) | 29.95 | a |
| GRAFIX 24-PIN UNIVERSAL PRINTER DRIVER | 14.95 | e |
| HARDBACK + FINDER HARD DISK MANAGER | 49.95 | c |
| IDIS INTELLIGENT DISASSEMBLER | 24.95 | a |
| IDIS SPECIAL EDITION | 39.95 | d |
| LIGHTNING ACCELERATOR | 24.95 | a |
| LIGHTNING SPECIAL EDITION ACCELERATOR | 49.95 | h |
| LIGHTNING SPECIAL EDITION FOR GOLD CARD | 39.95 | a |
| MAILMERGE DATABASE SYSTEM | 19.95 | a |
| MEDIA MANAGER SPECIAL EDITION | 49.95 | d |
| MEDIA MANAGER | 24.95 | a |
| MEGA DICTIONARY | 29.95 | g |
| MICROBRIDGE | 39.95 | d |
| NAMES AND ADDRESSES DATABASE SYSTEM | 19.95 | d |
| PAYROLL SYSTEM | 49.95 | c |
| PC CONQUEROR GOLD SPECIAL EDITION WITH DR-DOS v6.0 | 179.95 | g |
| PC CONQUEROR GOLD SPECIAL EDITION | 99.95 | g |
| PC CONQUEROR WITH DR-DOS v6.0 | 139.95 | e |
| PC CONQUEROR | 59.95 | e |
| PEDIT PRINTER DRIVER FOR XCHANGE | 19.95 | a |
| PERFECTION PLUS SPECIAL EDITION WITH SPELLCHECKER | 139.95 | c |
| PERFECTION PLUS WITH SPELLCHECKER | 99.95 | e |
| PERFECTION SPECIAL EDITION WORD PROCESSOR | 99.95 | c |
| PERFECTION WORD PROCESSOR | 59.95 | d |
| PERFECT POINTER TOOLS | 29.95 | a |
| PROFESSIONAL ASTROLOGER WITH ASTRONOMER | 69.95 | a |
| PROFESSIONAL ASTROLOGER | 59.95 | a |
| PROFESSIONAL ASTRONOMER | 29.95 | f |
| PROFESSIONAL PUBLISHER TOOLBOXES (PARTS ONE & TWO) | 49.95 | e |
| PROFESSIONAL PUBLISHER TOOLBOX PART ONE | 29.95 | e |
| PROFESSIONAL PUBLISHER TOOLBOX PART TWO | 29.95 | e |
| PROFESSIONAL PUBLISHER | 89.95 | c |
| QFLICK CARD INDEX SYSTEM | 29.95 | a |
| QKICK FRONT END SYSTEM | 24.95 | a |
| QMATHS MATHEMATICAL SYSTEM PART ONE | 69.95 | c |
| QMATHS MATHEMATICAL SYSTEM PART TWO | 59.95 | c |
| QMATHS MATHEMATICAL SYSTEM (PARTS ONE & TWO) | 99.95 | c |
| QMON MACHINE CODE MONITOR v2.05 | 39.95 | a |
| RECOVER ARCHIVE UTILITY | 19.95 | a |
| REVERSI GAME | 9.95 | f |
| SEDIT + SCREENPRINT ARCHIVE UTILITIES | 39.95 | a |
| SOLUTION PC EMULATOR WITH DR-DOS v6.0 | 109.95 | e |
| SOLUTION PC EMULATOR | 29.95 | e |
| SPELLCHECKER WITH DICTIONARIES | 49.95 | d |
| SUCCESS CP/M EMULATOR | 49.95 | b |
| SUPERBASIC MONITOR | 24.95 | a |
| SUPERFORTH COMPILER WITH REVERSI | 39.95 | a |
| SUPER ASTROLOGER | 24.95 | f |
| SUPER BACKGAMMON GAME | 9.95 | f |
| SUPER SPRITE GENERATOR | 29.95 | a |
| TOOLKIT III WITH ROM | 49.95 | h |
| TOOLKIT III | 29.95 | a |
| TRANSFER UTILITY SPECIAL EDITION | 29.95 | b |
| TRANSFER UTILITY | 9.95 | b |
| TURBO BASIC COMPILER + TOOLKIT | 79.95 | a |
| ULTRAPRINT SCREEN DUMP UTILITY | 19.95 | a |
| XREF SUPERBASIC PROGRAM ANALYSER | 29.95 | a |

NEW!

*KEY>>>>>>> Available either on cartridge or disk* a
*Available only on disk* b
*Minimum 512K exp:only available on disk* c
*Minimum 256K exp:either cartridge or disk* d
*Minimum 256K exp:only available on disk* e
*Available only on cartridge* f
*Minimum 1.5Mb RAM:only available on disk* g
*As well as cartridge or disk, you get a ROM* h

VISA

Send to: **DIGITAL PRECISION LTD, 222 THE AVENUE, CHINGFORD, LONDON E4 9SE** or phone **081-527 5493**
Please rush me: .....
Name: ..... Address: .....
..... Postcode: .....
Encl: CHEQUE/VISA/ACCESS/MASTERCARD/PO/MO/CASH for £ .....
Card No: ..... Expiry: ..... Signed: .....
Delete as appropriate: (QL/THOR/STQL)(MDV/3.5"/5.25")(Disk=360/720/1440/2880/6400 sectors; nb 1 sector=0.5K)(If 720 sectors: SSDD/DSSD)

DIGITAL PRECISION LTD

# LONELY JOKER

**INFORMATION**
**Program: The Lonely Joker**
**Supplier: Jochen Merz Software, Im Stillen Winkel 12, 4100 Duisberg 11, Germany.**
**Price: £13.00**

Imagine that it's time to relax and kill a few moments. You want something to do, but it has to be enjoyable and not taxing. The last thing you want is a dose of heavy concentration. Well, here's something you will want - designed specially with you in mind.

*The Lonely Joker* is an excellent representation of the classic patience or Solitaire card games featuring brilliant graphics and fine gameplay. The Lonely Joker will let you play at your own speed, with no pressure, and will lull you into a true sense of calmness, satisfaction and even achievement.

There are no opponents to beat, hard thinking is unnecessary, and you have all the time in the world to make your move. Ease of use is superb, too, especially if you have a Qimi+ mouse, as theprogram runs under the Pointer Environment, although it is perfectly feasible to use the keyboard.

## Suspense!

But don't get the impression that it lacks excitement. You can be progressing well, with cards present in the right place at the right time, when suddenly you reach an impasse. There seem to be no suitable cards in the deck. You think you're stuck. And then it all clicks into place. You move a sequence from one rank to another. That makes way for a card from the deck, and suddenly you're back in business. You go on to a rip-roaring finish. That's where the sense of achievement comes in.

## Four games

You have a choice of three games (or four if you count the two versions of Echelon).

*Echelon* requires you to build four consecutively-numbered suit foundations either directly from the deck, or via a series of ranks which you build up with consecutive numbers but alternating colours. The program defaults to this game, although you can configure it to start off with any one you want.

The next game is *Napoleon,* and is linked to the great emperor and strategist himself. The object of the game is similar to that of Echelon, but with a totally different layout and the facility to place cards in a temporary store in an effort to make way for more strategically-important cards. A number of other restrictions make this one of the hardest of the games to finish.

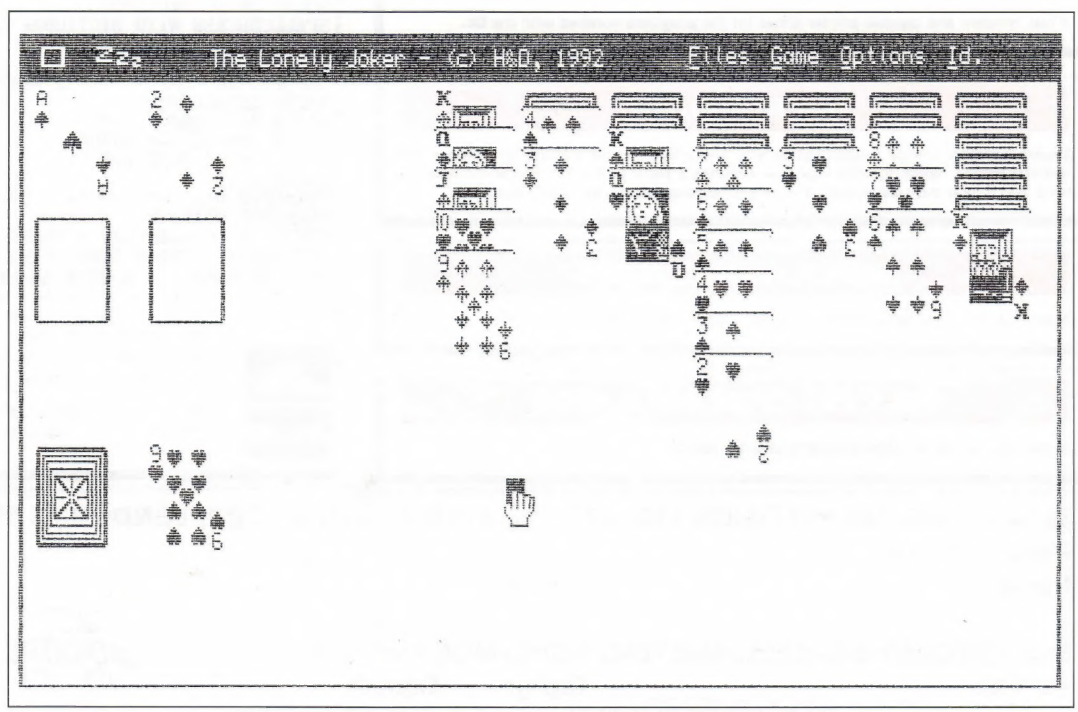
*Cascade* is different again. Here you are required to place cards in an ascending and/or descending order (this can alternate) regardless of suit. The skill needed is the ability to choose the order based on the availability of cards and possibility of finishing.

## Delegation

It sounds easy, but it's not. While it's not hard work, you do need just the right among of skill and strategic thinking. But, with foresight to cater for the lazy among you, if the program thinks it can finish the game for you, it will ask you if you want it to do so.

Other nice touches include an option to note the time taken to finish a game, and a choice of green (the default) or black boards. For the more enterprising among you, it's even possible to make the game more difficult by imposing a restriction on the number of card turns permitted.

I thoroughly recommend this software, and it comes at a very reasonable price.




**Henry Orlowski relaxes (alone) with the Joker**

**Eidersoft Ice rom** (windows,icons,pointers)front end ,choice multitasking & mailmerge m'drive cartridge,plus 128k RAM extension in TECHNOLOGY RESEARCH Delta discdrive/printer interface. All manuals and documentation included.Complete package £60. Telephone 0732 355341

**PC Conqueror** original £65 QL computer vers JM £60 super toolkit 2 EPROM cartridge £18 Centronics printer interface £18 Philips green Hi-res monitor BM7502 £70.Telephone 0233 626713 (Kent)

**QL £50 green** monitor £30, real time digitizer £50,Twin disk drive £50, software:digital S.E. £25, editor S.E. £25,touch typist £6,assembler £12.Telephone 021449 8041

**Unused Penman** plotter suitable for use with QL,PC,BBC and RM computers. Up to A2 size. Bargain at only £120. Tel:0449 767130

**QL 640k twin** 3-1/2" drives, mouse, minerva, cased seperate keyboard £150. Tonto with monitor £100. Qram,painter,eye Q, CPM emulator, sprite generator,MDVS,games,books ,mags. Tel: Jeff 0908 233008

**QL Minerva Rom,Trump** Card 3-1/2" D/D,Miracle centronics (x2) Tandata modem,serial printer, 6 years QLWorld, Quanta mags, books, software, b/w television extra MDV. Sensible offers please. Tel: 0977 515428

**QL.JM Trump** card 512k TR.128k interface books, programs £60 Tel: 081 998 8018

**640k JS QL twin** 3-1/2" drives housed printer, monitor, schon keyboard, Tandata modem, eidersoft mouse. V2.35 Psion suite turbo Quill taskmaster spellbound speedscreen £220 ono. buyer collects. Tel: 0765 640331

**QL unexpanded** AH version PSU Psion software manual etc. Any offers Miracle Systems QL modem £20. Phone: John 0689 855444

**JS QL, Metal** case, PC keyboard and I/face, twin 3-1/2" d/drives and OPD m/drives, some software, new colour monitor £250 ono. Also Tandata modem 3, CQVI Digitiser and more. Tel: (Derby) 0332 677746

**Dual disk drive** £75 toolkit III £15 success £25 superforth £25 Digital C £20 Qload £10 QLiberator MDV £12. Better Basic £12. Gold card £200 HD/DD drive £100. Phone: Paul 0795 535928

---



# CLASSIFIED

Please send coupon to: Advertising Department Sinclair/QL World, The Blue Barn, Wootton, Woodstock, Oxon. OX7 1HA. Please place this Classified Advertisement in the next issue of Sincair/QL world.

NAME .....

ADDRESS .....

.....

DAYTIME TELEPHONE NO .....

I enclose a cheque/P.O. to the value of £17.65 (inc. VAT) for a lineage advertisement of no more than 30 words. Please make cheques payable to: Arcwind Ltd.

| | | | | | |
|---|---|---|---|---|---|
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |

# DJC

# Dilwyn Jones Computing

41 Bro Emrys, Tal-y-Bont,
Bangor, Gwynedd LL57 3YT U.K.
Tel: Bangor (0248) 354023

## GRAPHICS

THE PAINTER V4.04 £25.00
*[F 512k] 100% machine code pointer driven art and graphics program, by PROGS of Belgium. Can be mouse controlled. Patterns, fills, size changes, screen dumps, all sorts of facilities.*

THE CLIPART £12.00
*[F 128k] 3 disks packed full of screen pictures for use in desktop publishing and graphics programs. Ideal for use with The Painter or Image Processor or Page Designer, for example!*

PICTORIAL INDEX FOR THE CLIPART £4.00
New printed index for the clipart with screen dumps of all pictures for easy selection.

QRACTAL £20.00
*[F 512k] Machine code fractals program, able to display Julia or Mandelbrot sets.*

QRACTAL SCREENS £5.00
*[F 128k] Sample screen pictures made with QRactal.*

IMAGE PROCESSOR 2 £15.00
*[F 512k] Picture editing and creation system, image enhancement, mode conversion, touch up pictures from other sources. Can be used to tidy up screens from digitisers, for example, or just as an art and graphics program.*

PD2 CLIPART £10.00
*[F 128k] Clipart from Page Designer 2, supplied in mode 4 screens format for use with any graphics program.*

SCREEN SNATCHER £10.00
*[F M 128k] Grab screen displays from within other programs for use in your own programs!*

TEXT 'N' GRAPHIX £20.00
*[F 256k] Print text files containing screen dumps, mix text and graphics in same document.*

TRANS24 £10.00
*[F M 128k] Translate 9 pin graphics output from most programs without a 24 pin printer driver to 24 pin.*

## TEXT

QTYP 2 £29.95
*[F 512k] Typing and spelling checker, which uses pointer environment. 40,000 word English dictionary plus French and German dictionaries. Dictionary editor included to add your own words. SuperBASIC interface allows you to write your own BASIC programs which use QTYP.*

BIBLE TEXT DISKS, PLAIN TEXT FORMAT £20.00
BIBLE TEXT DISKS, QUILL FILE FORMAT £20.00
*[F 512k] Text of the King James Bible on disks.*

SPELLBOUND £30.00
*[F M 384k] Spelling checker which checks your typing as you type! 30,000 word dictionary, expandable.*

SPELLBOUND SPECIAL EDITION £50.00
*[F 512k] Improved version with larger 50,000 word dictionary (expandable). Check spelling as you type or check existing text files.*

UPGRADE TO SPECIAL EDITION £30.00
*Please return master disk/cartridge when upgrading.*

QUICK POSTERS £10.00
*[F M 128k] Text poster maker for use with Star LC, NL and XB printers. Current version works in pointer environment, upgrade from old version £3.00*

## DATABASES

ADDRESS BOOK & LABEL PRINTER £15.00
*[F M R 384k] Store names and addresses and print them out on a variety of label sizes, print telephone lists, etc.*

DATA DESIGN £50.00
*[F 512k] A database which can be programmed from BASIC or machine code. Runs in pointer environment which is supplied with the program. Fast and versatile.*

FLASHBACK £25.00
*[F M 256k] Fast database written in machine code, not programmable.*

FLASHBACK SPECIAL EDITION £40.00
*[F 256k] Enhanced version of Flashback, with Report Generator, etc. NB Please advise if to be used on a Gold Card, since Reporter Generator requires modification.*

QL GENEALOGIST SECOND EDITION £30.00
*[F 384k] Family trees and family history program, one of our best selling programs!*

UPGRADE TO SECOND EDITION £12.00
*Please return master disk with upgrade order.*

BUDGET 128K GENEALOGIST £12.00
*[F M 128k] A cut down version for unexpanded machines, still superb value for money.*

DISK INDEXER £12.00
*[F M 128k] Make a database of the contents of your disks, so that you can search for files, list them, and generally tidy up your disks and cartridges!*

DBEASY £15.00
*[F 512k] Database front end for Archive and a suite of programs called Chaos Busters!*

DBPROGS £15.00
*[F 512k] A collection of Archive utilities and text files to help you to learn to program Archive.*

## D.T.P.

PAGE DESIGNER 2 PLUS £40.00
*[F 512k] QL Desktop publishing program, mix text and graphics, print pages, use graphics from other programs. Prints on most printers. Ask for information. Sadly, it has been badly delayed, so you should check with DJC first to see if available yet before ordering.*

UPGRADE TO PD2 PLUS £20.00
*Send proof of purchase of old Page Designer 2 with upgrade order. See note above re. delays.*

## QL HARDWARE

MINI PROCESS CONTROLLER £59.95
*Relay switched outputs, controlled via QL serial ports.*

SOFTWARE TOOLKIT FOR MPC £9.95
*[F 128k] BASIC extensions to simplify writing programs to control the MPC.*

NETWORK PROVER £3.50
*A small box which plugs between networked QLs to give a visual indication of transmission over the network.*

QPOWER REGULATOR £24.95

## OTHER QL SOFTWARE

QPAC1 £19.95
*[F 512k] Pointer environment utilities, including calculator, calendar, alarm, typewriter, clock and system monitor utility. Pointer environment included. Quality QL software by Tony Tebby.*

QPAC2 £39.95
*[F 512k] If you want to make full use of multitasking on your QL, this package is a must. Files maintenance menu, hotkeys, "things", channels, pick, etc. Extensive documentation and a tutorial is supplied. Another Tony Tebby package.*

QTOP £29.50
*[F 256k TK2] User front end for QDOS by Cowo Electronic of Switzerland. Desktop system, file handling, job control, program startup menus, with 5 desktop accessory programs.*

PRINTERMASTER £20.00
*[F M 128k] Printer control utility - select fonts, set margins, page length and much more. Printermaster puts you in control of your printer. Not a graphics program, not a printer driver creator, just a database of printer functions with the ability to send control codes to the printer quickly and easily without having to reach for your manual. Files for several makes of printers supplied, or create your own or modify those supplied quickly and easily. In constant use at DJC, that's what I think of it!*

HOME BUDGET £20.00
*[F M 128k] Domestic bills and accounts program.*

REMIND-ME £12.00
*[F M 128k] Dates reminder program. Quick and easy to use.*

REMIND-ME PLUS £20.00
*[F M 128k] Enhanced to allow more dates to be scheduled, longer event descriptions.*

SCREEN ECONOMISER £10.00
*[F M 128k] Turns off the QL display after a set number of minutes during which QL has not been used to protect the screen.*

SLOWGOLD £5.00
*[F 128k] Slowdown routine and control panel for software which runs too fast on a Gold Card or indeed on any QL system.*

TASKMASTER £25.00
*[F M 384k] Task switching utility. Enables you to conveniently switch between several programs in memory. Calculator, notepad and file handling utility included. Ideal if you want a simple to use system to let you have several programs in the computer at once and to switch between them at the press of a couple of keys. Can even save and restore the display of each program automatically.*

DISK LABELLER £10.00
*[F 256k] Prints neat labels for your floppy disks, listing the filenames from the disk in columns in small print on the label. Essential starting point to tidying up your disk collection. We have sold a great deal of copies of this useful little program - have you got yours yet?*

THE CAT £5.00
*[F M 128k] List filenames from a disk or cartridge in columns to the screen or printer. The list can be sorted if required. Useful and convenient utility. Installs as a BASIC extension then just type CAT for a list of files.*

THE SMALL PRINT! POSTAGE AND PACKING CHARGES: Software is sent post-free to UK addresses. To other countries, please add £1.00 per program for postage and packing (sent by airmail where possible). PRICES: All prices are shown in UK Pounds Sterling. PAYMENT: We can accept payment by cheque (in UK Pounds Sterling currency only, please) drawn on UK branch of a bank or building society, by Eurocheque with card number written on the back, Postal Order, or by these credit cards: VISA, ACCESS, MASTERCARD, EUROCARD or by CONNECT card. Please state the card type, number, expiry date, your address, and sign orders sent by post. We can also accept orders paid by credit card over the telephone. There is an answering machine for when I am unable to answer in person so that I can call you back later. Goods remain the property of DJC until paid for in full. PLEASE STATE IF YOU REQUIRE SOFTWARE ON 3.5 OR 5.25 INCH DISKS OR MICRODRIVE CARTRIDGE.